

# Masterarbeit

## Entwicklung eines semi-automatischen Workflows zur Ableitung ikonographischer Kartenzeichen

eingereicht von  
geboren am

**Ronny Techt**  
12.06.1986 in Dresden

zum Erlangen des Hochschulgrades  
Master of Science (M.Sc.)

Tag der Einreichung 30.04.2020

Betreuer

Prof. Dipl.-Phys. Dr.-Ing. habil. Dirk Burghardt  
Institut für Kartographie, TU Dresden



# Selbstständigkeitserklärung

Hiermit erkläre ich, dass die von mir am heutigen Tage der Prüfungskommission der Fachrichtung Geowissenschaften eingereichte Abschlussarbeit zum Thema

## **„Entwicklung eines semi-automatischen Workflows zur Ableitung ikonographischer Kartenzeichen“**

vollkommen selbstständig und nur unter Benutzung der in der Arbeit angegebenen Quellen und Hilfsmittel angefertigt wurde. Wörtlich oder sinngemäß übernommene Zitate wurden als solche kenntlich gemacht.

Dresden, den 30.04.2020

---

Unterschrift



# Inhaltsverzeichnis

Selbstständigkeitserklärung.....	III
Inhaltsverzeichnis .....	5
Abbildungsverzeichnis .....	7
Tabellenverzeichnis.....	8
Abkürzungsverzeichnis .....	9
1    Einleitung.....	10
1.1    Motivation.....	10
1.2    Gliederung der Arbeit.....	10
2    Literaturstudium .....	11
2.1    Computergrafik.....	11
2.2    Non-Photorealistic Rendering.....	11
2.3    Neural Style Transfer.....	14
2.3.1    Einleitung.....	14
2.3.2    Convolutional Neural Network .....	15
2.3.3    Beschreibung des Algorithmus .....	17
3    Methodik .....	19
3.1    Technische Komponenten.....	19
3.2    Kriterien der Bildauswahl .....	19
3.3    Workflow „Ölmalerei“ .....	21
3.4    Workflow „Tuschezeichnung“ .....	22
3.5    Workflow „Silhouette“ .....	22
4    Praktischer Teil .....	23
4.1    Konkrete Umsetzung.....	23
4.1.1    Workflow „Ölmalerei“ .....	24
4.1.2    Workflow „Tuschezeichnung“ .....	32
4.1.3    Workflow „Silhouette“ .....	32
4.2    Implementierung eines Automatisierungsprozesses.....	35
4.3    Anwendung: Karte Dresden .....	39
4.4    Neural Style Transfer.....	43
4.4.1    Online-Anwendungen .....	43

4.4.2	Offline-Implementierung .....	45
5	Diskussion.....	51
5.1	Resultate .....	51
5.1.1	Bildverarbeitung.....	51
5.1.2	Neural Style Transfer .....	51
5.2	Ausblick.....	52
6	Zusammenfassung .....	52
	Literaturverzeichnis .....	53

## Abbildungsverzeichnis

Abbildung 1: Stift und Tinte Zeichnung von (Winkenbach & Salesin, 1994) .....	12
Abbildung 2: Abstrakte Weltkarte gerendert mit sketchy 'marker pen'-Stil .....	12
Abbildung 3: Watercolor Style von Stamen (Ausschnitt Dresden) .....	13
Abbildung 4: Bilddarstellungen in einem Convolutional Neural Network (CNN) .....	16
Abbildung 5: Aufbau Convolutional Neural Network VGG-19 .....	16
Abbildung 6: Style Transfer Algorithmus .....	18
Abbildung 7: User Interface G'MIC für GIMP .....	20
Abbildung 8: Arbeitsschritte Workflow "Ölmalerei" .....	21
Abbildung 9: Ausgangsbild Frauenkirche Dresden .....	23
Abbildung 10: Katholische Hofkirche Dresden .....	24
Abbildung 11: Auswahlwerkzeuge in Gimp .....	24
Abbildung 12: Auswahl mit "Magnetischer Schere" .....	25
Abbildung 13: Ausgangsbilder nach Freistellung und Skalierung .....	25
Abbildung 14: Ergebnis nach Einsatz Normalize Brightness und Retinex .....	26
Abbildung 15: Ergebnis nach Schärfung von Makrostrukturen .....	27
Abbildung 16: Ergebnis Filter Posterize .....	27
Abbildung 17: Ebenen-Dialog in GIMP .....	28
Abbildung 18: Ergebnis nach Engrave und Anpassung Ebenenmodus .....	29
Abbildung 19: Ergebnis Filter Sketch .....	29
Abbildung 20: Extrahierte Faser, Farben entsättigt .....	30
Abbildung 21: Endergebnis Workflow Ölmalerei .....	31
Abbildung 22: Endergebnis mit Konturlinie und Schatten .....	31
Abbildung 23: Vergleich Ausgangsbild und Ergebnis Ölmalerei (verkleinert) .....	32
Abbildung 24: Entsättigung der Farben .....	33
Abbildung 25: Ergebnis Morphological Filter .....	33
Abbildung 26: Ergebnis Filter Stamp .....	34
Abbildung 27: Endergebnis Workflow Tuschezeichnung .....	34
Abbildung 28: Vergleich Ausgangsbild, Endergebnis Tuschezeichnung mit Konturlinie u. Schatten (verkleinert) .....	34
Abbildung 29: Vergleich Ausgangsbild und Endergebnis Silhouette mit Konturlinie und Schatten (verkleinert) .....	35
Abbildung 30: Ausschnitt Karte Dresden mit Ölmalerei-Signaturen (Maßstab 1:8.000) .....	40
Abbildung 31: Ausschnitt Karte Dresden mit Tuschezeichnung-Signaturen (Maßstab 1:16.000) ....	41
Abbildung 32: Ausschnitt Karte Dresden mit Silhouette-Signaturen (Maßstab 1:32.000) .....	42
Abbildung 33: Showcase Webseite von Deep Art Effects Developer .....	43
Abbildung 34: Beispiele NST über Webseite von Deep Art Effects Developer .....	44
Abbildung 35: Ausgangsbild Frauenkirche (links) und Ergebnis Workflow Ölmalerei (rechts) .....	45
Abbildung 36: NST Basemap .....	46
Abbildung 37: Ergebnis NST - Karte Deutschland mit Icons der Sehenswürdigkeiten .....	50
Abbildung 38: Kulturpalast Dresden - NST mit Ölmalereistil .....	52

## Tabellenverzeichnis

Tabelle 1: Parameter Filter Normalize Brightness.....	26
Tabelle 2: Parameter Filter Engrave .....	28
Tabelle 3: Kriterien der Reproduzierbarkeit .....	35
Tabelle 4: Inhalt Datei user.gmic.....	37
Tabelle 5: Fortsetzung Inhalt Datei user.gmic.....	38
Tabelle 6: Übersicht der wichtigsten Parameter und verwendete Werte .....	46
Tabelle 7: Übersicht der Ergebnisse des NST .....	49



## Abkürzungsverzeichnis

CC.....	Creative Commons
CeCILL.....	CEA CNRS INRIA Logiciel Libre
CLI.....	Command-Line Interface
CNN .....	Convolutional Neural Networks
EBR .....	Example-Based Rendering
G'MIC .....	GREYC's Magic for Image Computing
GIMP.....	GNU Image Manipulation Program
IB-AR.....	Image-based Artistic Rendering
JPG .....	Joint Photographic Experts Group
NPR.....	Non-Photorealistic Rendering
NST .....	Neural Style Transfer
OSM.....	Open Street Map
PNG .....	Portable Network Graphics
ReLU .....	Rectified Linear Unit

# 1 Einleitung

## 1.1 Motivation

Die Rolle der Karte ist bis heute allgegenwärtig. Sie stellt einen wichtigen Teil zur Sicht und Interpretation unserer Welt dar. Lediglich die Formen der Präsentation haben sich im Laufe der Zeit weiterentwickelt. War es im Mittelalter die handgezeichnete Karte oder in der Neuzeit der gedruckte Stadtplan, so leitet uns heute oft die Navigations-App per Smartphone von A nach B. Karten dienen der Orientierung. Sie bilden topographische Informationen ab, werden aber auch zur Darstellung z.B. politischer oder gesellschaftlicher Themen benutzt. Entwicklungen zeigen, dass dabei der Kreativität der Gestaltung keine Grenzen gesetzt sind, wie die Sammlung von (Desclaux-Salachas, 2017) sehr gut zeigt. Die Kartographie ist eine interdisziplinäre Wissenschaft. Es begegnen sich Kunst und Wissenschaft. Aus dieser Motivation heraus ist diese Arbeit entstanden.

## 1.2 Gliederung der Arbeit

Die vorliegende Arbeit beschäftigt sich mit der Erstellung von bildhaften Signaturen prominenter Gebäude. Dabei sollen Bildaufnahmen mittels Bildverarbeitungsmethoden in eine künstlerische Darstellung überführt werden. Ziel dabei ist es, einen automatisierten Workflow für das Vorgehen zu entwickeln und die Ergebnisse in einer Karte darzustellen.

In Kapitel 2 wird ein kurzer Literatureinblick in die Thematik gegeben. Außerdem wird die Methode des Neural Style Transfer beschrieben.

Im Rahmen der Arbeit sind drei Workflows entwickelt worden. Kapitel 3 beschreibt das methodische Vorgehen und die eingesetzten technischen Komponenten. Anschließend wird im Kapitel 4 die konkrete Umsetzung beschrieben. Hierbei wird das genaue Vorgehen anhand diverser Beispiele aufgezeigt. Die Entwicklung eines Automatisierungsprozesses, sowie die Anwendung in einer Kartendarstellung runden das Kapitel ab. Im Anschluss wird eine praktische Umsetzung des Neural Style Transfer vorgestellt. Kapitel 5 diskutiert kurz die Ergebnisse.

## 2 Literaturstudium

### 2.1 Computergrafik

Der Begriff Computergrafik beschreibt jede Verwendung von Computern zum Erstellen und Bearbeiten von Bildern. Das Themengebiet lässt sich grundlegend in drei Bereiche gliedern. Die Modellierung befasst sich mit der mathematischen Spezifikation von Form- und Erscheinungseigenschaften, die auf dem Computer gespeichert werden kann. Das Rendering, das die Erstellung von Bildern aus Modellen übernimmt und die Animation, bei der durch Bildsequenzen eine Illusion von Bewegung erzeugt wird. Darüber hinaus sind Bereiche wie Benutzerinteraktion, Virtuelle Realität, Visualisierung, Bildverarbeitung, 3D-Scannen und Computerfotografie verwandte Bereiche. Anwendung finden die Technologien u.a. in den Bereichen Videospiele, visuelle Effekte, Animationsfilme oder Informationsvisualisierung. (Shirley, et al., 2009)

### 2.2 Non-Photorealistic Rendering

Nicht-fotorealistische Computergrafiken beschreiben einen Bereich, der sich mit der Erzeugung von Bildern befasst, die scheinbar handgemacht sind. Diese Bilder zeichnen sich eher durch Zufälligkeit, Mehrdeutigkeit oder Willkür als durch Vollständigkeit und Kohärenz aus. Obwohl es alle Aspekte der Computergrafik abdeckt, handelt es sich hauptsächlich um den Rendering-Prozess, der als Non-Photorealistic Rendering (NPR) bezeichnet wird. (Strothotte & Schlechtweg, 2002)

Es stellt eine hochaktive Forschungsdomäne der Computergrafik dar, die sich mit dem Ausdruck, der Erkennung und der Kommunikation komplexer Bildinhalte mittels Informationsabstraktion und Hervorhebung befasst (DeCarlo & Santella, 2002), (Gooch, 2010), (Hertzmann, 2010), (Lansdown & Schofield, 1995). Insbesondere erfreut sich das bildbasierte künstlerische Rendering (IB-AR) wachsender Beliebtheit beim mobilen expressiven Rendering (Dev, 2013), (Winnemöller, 2013), um die Attraktivität traditioneller künstlerischer Stile und Medien für die visuelle Kommunikation zu simulieren (Kyprianidis, Collomosse, Wang, & Isenberg, 2013), (Rosin & Collomosse, 2013) wie Bleistift, Feder, Ölfarbe und Aquarell. Klassische IB-AR-Techniken modellieren typischerweise die Designaspekte, die mit diesen künstlerischen Stilen verbunden sind, d.h. um die Glättung und Konturhervorhebung der Bildfilterung, die Approximation des Bildinhalts über Rendering-Grundelemente, wie z. B. Pinselstriche und Tupfen oder eine Bildsegmentierung zu steuern. Ein allgemeinerer Ansatz wurde durch beispielbasiertes Rendering (EBR) eingeführt, bei dem maschinelles Lernen oder statistische Modelle verwendet werden, um Merkmale künstlerischer Stile anhand visueller Beispiele nachzuahmen (Kyprianidis, Collomosse, Wang, & Isenberg, 2013). (Semmo, Isenberg, & Döllner, 2017)

Eine der bekanntesten handgezeichneten Rendering-Techniken in NPR ist wahrscheinlich das Zeichnen mit Stift und Tinte (Pen-and-Ink-Drawing). Dieser Ansatz wurde erstmals von (Winkenbach & Salesin, 1994) eingeführt und definiert Algorithmen und Techniken, um ein bestimmtes künstlerisches Styling zu erreichen. Es werden „Strich-Texturen“ zur Erzeugung von Ton und Textur sowie „Schraffur mit kontrollierter Dichte“ für Freiformflächen verwendet (Abbildung 1). Dieses Rendering stellt eine Vereinfachung des ursprünglichen Bildinhalts dar und wird in technischen Zeichnungen häufig bevorzugt.

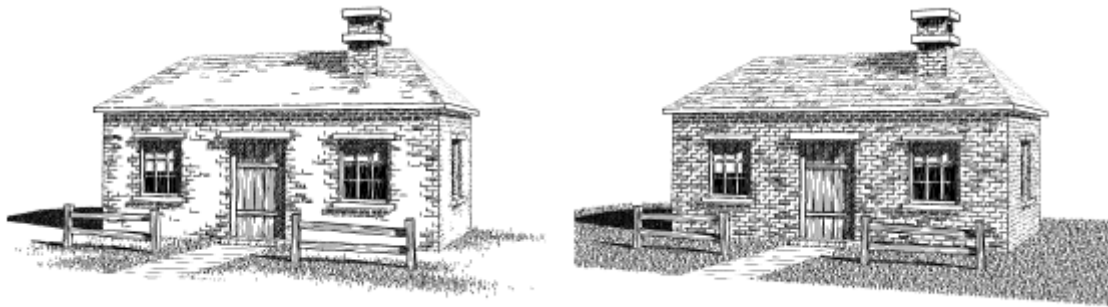


Abbildung 1: Stift und Tinte Zeichnung von (Winkenbach & Salesin, 1994)

Ein Framework für eine vektorbasierte handgezeichnete Rendering-Technik wurde von (Wood, et al., 2012) entwickelt und evaluiert. Zur Visualisierung von Informationen ist ein skizzenhafter Rendering-Stil als Bibliothek für die Programmiersprache Processing<sup>1</sup> implementiert worden. Diese ist auf die Einsatzbereiche Grafik, Simulation und Animation spezialisiert. Der Stil mit dem Namen Handy eignet sich zum Zeichnen von Grundelementen wie Linien, Polygonen und Ellipsen. Es soll die Idee der Ungenauigkeit, die Ästhetik und die Qualitäten einer Visualisierung verbessern. Dies ist von besonderem Interesse für das schnelle digitale Prototyping von Schnittstellen, bei dem es wichtig ist, darauf hinzuweisen, dass der Entwurfsprozess noch nicht abgeschlossen ist. Darüber hinaus wird die Skizzenhaftigkeit als visuelle Variable betrachtet, die geändert werden kann und Informationen enthält. Schließlich kann das handgezeichnete Rendering ein Werkzeug sein, um diesen Stil von gewöhnlichen computergenerierten Designs abzuheben. Ein Beispiel hierfür ist in Abbildung 2 dargestellt. (Wood, et al., 2012)



Abbildung 2: Abstrakte Weltkarte gerendert mit sketchy 'marker pen'-Stil (Wood, et al., 2012)

---

<sup>1</sup> <https://processing.org/>

Die bekannteste rasterbasierte Webkarte ist die von Stamen<sup>2</sup> (Abbildung 3). Es handelt sich um eine OpenStreetMap Webkarte im Aquarellstil (watercolor style). Obwohl die Ausgangsdaten Vektordaten sind, werden alle Stiling-Arbeiten im Rasterbereich durchgeführt. Der genaue Erzeugungsprozess ist schrittweise auf dem internen Blog<sup>3</sup> beschrieben. Zunächst werden einige Bilder in Mapnik aus den Vektordaten gerendert und anschließend entsprechende Stilingoperatoren, wie z.B. Bildteilung und Gaußsche Unschärfe, der Reihe nach angewendet.



Abbildung 3: Watercolor Style von Stamen (Ausschnitt Dresden)

---

<sup>2</sup> <http://maps.stamen.com/#watercolor/12/37.7706/-122.3782>

<sup>3</sup> <https://hi.stamen.com/watercolor-process-3dd5135861fe>

## 2.3 Neural Style Transfer

Die Malerei ist eine beliebte Kunstform. Der Mensch ist jeher von der Kunst des Malens angezogen und inspiriert worden. Die Nachahmung eines bestimmten Stils eines Kunstwerkes, z.B. Vincent van Gogh's „Sternennacht“ setzt dabei besondere Fähigkeiten eines gut ausgebildeten Künstlers und viel Zeit voraus (Jing, et al., 2018).

### 2.3.1 Einleitung

Das Rendern dieses Prozesses, des semantischen Inhalts eines Bildes in verschiedene Stile, ist eine schwierige Bildverarbeitungsaufgabe. 2015 veröffentlichen (Gatys, Ecker, & Bethge, 2015) A Neural Algorithm of Artistic Style. Eine wegweisende Arbeit, die die Erzeugung künstlerischer Bilder durch Trennung und Rekombination von Bildinhalt und Stil beschreibt. Dabei kommen sog. Convolutional Neural Networks (CNNs) zum Einsatz. Dieser Prozess zum Rendern eines Inhaltsbilds in verschiedene Stile wird als Neural Style Transfer (NST) bezeichnet. NST ist seitdem sowohl in der akademischen Literatur als auch in industriellen Anwendungen zu einem Trendthema geworden. Es wird zunehmend beachtet. Außerdem werden verschiedene Ansätze behandelt, den ursprünglichen NST-Algorithmus entweder zu verbessern oder zu erweitern (Jing, et al., 2018).

Die Übertragung eines Stils von einem Bild auf ein anderes kann als Problem der Texturübertragung angesehen werden. Dabei besteht das Ziel darin, die Textur aus einem Quellbild zu synthetisieren. Die Textursynthese verläuft eingeschränkt, um den semantischen Inhalt eines Zielbildes zu erhalten. Die meisten früheren Texturübertragungsalgorithmen stützen sich auf nichtparametrische Methoden zur Textursynthese. Obwohl diese Algorithmen bemerkenswerte Ergebnisse erzielen, unterliegen sie alle derselben Einschränkung. Sie verwenden nur low-level Bildmerkmale des Zielbilds für die Texturübertragung. Ein Stilübertragungsalgorithmus sollte jedoch in der Lage sein, den semantischen Bildinhalt aus dem Zielbild, wie z. B. den Objekten und der allgemeinen Szenerie (high-level) zu extrahieren und eine Texturübertragungsprozedur durchzuführen, um den semantischen Inhalt des Zielbilds im Stil des Quellbildes zu rendern. Grundvoraussetzung ist daher, Bild Darstellungen zu finden, die die Variationen des semantischen Bildinhalts und des Stils, in dem er dargestellt wird, unabhängig voneinander modellieren. (Gatys, Ecker, & Bethge, 2016)

Die Fortschritte in der Computer Vision haben dazu geführt, dass Deep Convolutional Neural Networks per Lernprozess in der Lage sind semantische Informationen (high-level Merkmale) aus natürlichen Bildern zu extrahieren. Somit eignen sich Convolutional Neural Networks nicht nur für die Objekterkennung und Bildklassifizierung, sondern auch für die Stilübertragung. (Gatys, Ecker, & Bethge, 2016) zeigen in ihrer Arbeit, wie die von leistungsstarken CNNs gelernten generischen Merkmalsdarstellungen verwendet werden können, um den Inhalt und den Stil natürlicher Bilder unabhängig zu verarbeiten und zu manipulieren. Für die Durchführung der Bildstilübertragung führen sie einen Texturübertragungsalgorithmus ein, der ein Textur-Syntheseverfahren durch Merkmalsdarstellungen aus modernen CNNs erzwingt. Da das Texturmodell auch auf tieferen Bilddarstellungen (deep image representations) basiert, reduziert sich das Stilübertragungsverfahren auf ein Optimierungsproblem innerhalb eines einzelnen neuronalen Netzwerks. Neue Bilder werden durch Ausführen einer Vorbildsuche generiert, um Merkmalsdarstellungen von Beispielbildern abzugleichen. Der Stilübertragungsalgorithmus kombiniert ein parametrisches Texturmodell, das auf Convolutional Neural Networks basiert, mit einer Methode zum Invertieren ihrer Bilddarstellungen. (Gatys, Ecker, & Bethge, 2016)

### 2.3.2 Convolutional Neural Network

„Gefaltete Neuronale Netzwerke“ bestehen aus Schichten kleiner Recheneinheiten, die visuelle Informationen hierarchisch und vorwärtsgerichtet (feed-forward) verarbeiten (Abbildung 4). Jede Ebene von Einheiten kann als Sammlung von Bildfiltern verstanden werden, von denen jeder ein bestimmtes Merkmal aus dem Eingabebild extrahiert. Während die Anzahl der verschiedenen Filter entlang der Verarbeitungshierarchie zunimmt, wird die Größe der gefilterten Bilder durch einen Downsampling-Mechanismus (Max-Pooling oder Mittelwert-Pooling) verringert, was zu einer Verringerung der Datenmenge pro Layer des Netzwerks führt. Die Ausgabe eines bestimmten Layers besteht aus sog. Feature-Maps, also unterschiedlich gefilterten Versionen des Eingabebilds. Entlang der Verarbeitungshierarchie des Netzwerks wird das Eingabebild in Darstellungen umgewandelt, die zunehmend den tatsächlichen Inhalt des Bildes im Vergleich zu seinen detaillierten Pixelwerten enthalten. Die Informationen, die jeder Layer über das Eingabebild enthält, können direkt visualisiert werden, indem das Bild nur aus den Feature-Maps in diesem Layer rekonstruiert wird (Abbildung 4, Content Reconstructions). Höhere Ebenen im Netzwerk erfassen high-level Inhalte in Bezug auf Objekte und deren Anordnung im Eingabebild. Sie belegen jedoch nicht die genauen Pixelwerte der Rekonstruktion. Im Gegensatz dazu geben Rekonstruktionen aus den unteren Ebenen die genauen Pixelwerte des Originalbilds wieder. Die Feature-Antworten in höheren Schichten des Netzwerks werden als Inhaltsdarstellung (Content Representations) bezeichnet. (Gatys, Ecker, & Bethge, 2015)

Um die Stildarstellung eines Eingabebilds zu erhalten wird ein Feature-Space verwendet. Dieser Merkmalsraum basiert auf den Filterantworten in jeder Ebene des Netzwerks. Dieser besteht aus den Korrelationen zwischen den verschiedenen Filterantworten über die räumliche Ausdehnung der Feature-Maps. Durch Einbeziehen der Merkmalskorrelationen mehrerer Ebenen erhält man eine stationäre, mehrskalige Darstellung des Eingabebildes, die die Texturinformationen erfasst, jedoch nicht die globale Anordnung. Die Stilrekonstruktion eines Eingabebildes erfolgt aus Stildarstellungen (Style Representations), die auf verschiedenen Teilmengen von CNN-Ebenen basieren. In der Tat erzeugen Rekonstruktionen aus den Stilmerkmalen texturierte Versionen des Eingabebildes, die das allgemeine Erscheinungsbild in Bezug auf Farbe und lokalisierte Strukturen erfassen. Darüber hinaus nimmt die Größe und Komplexität lokaler Bildstrukturen aus dem Eingabebild entlang der Hierarchie zu, was durch die zunehmenden Empfangsfeldgrößen und die Komplexität der Merkmale erklärt werden kann. (Gatys, Ecker, & Bethge, 2015)

Eines der bekanntesten und für die Durchführung von Objekterkennung und -lokalisierung geschultes (Russakovsky, et al., 2015) CNN ist VGG-19. Das VGG-19-Netzwerk erschien 2014 in der Veröffentlichung von (Simonyan & Zisserman, 2015). Abbildung 5 zeigt den Aufbau. Das Netzwerk verwendet einen 3x3-Faltungskern und maximales Pooling. Es besteht aus 16 Convolutional-Layern. Zwischen diesen Faltungsebenen befinden sich immer im Wechsel insgesamt 5 Pooling-Layer. Schließlich gibt es noch drei Fully-connected Layer.



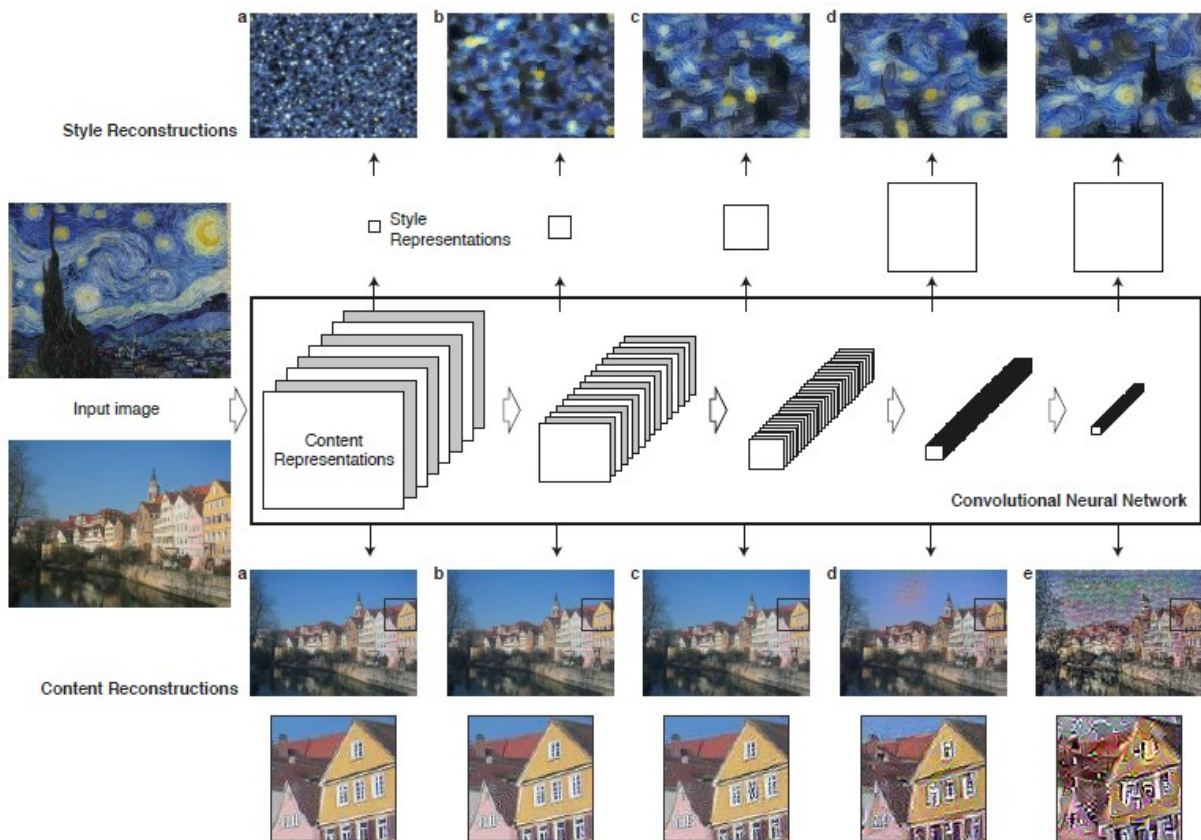


Abbildung 4: Bilddarstellungen in einem Convolutional Neural Network (CNN) (Gatys, Ecker, & Bethge, 2016)

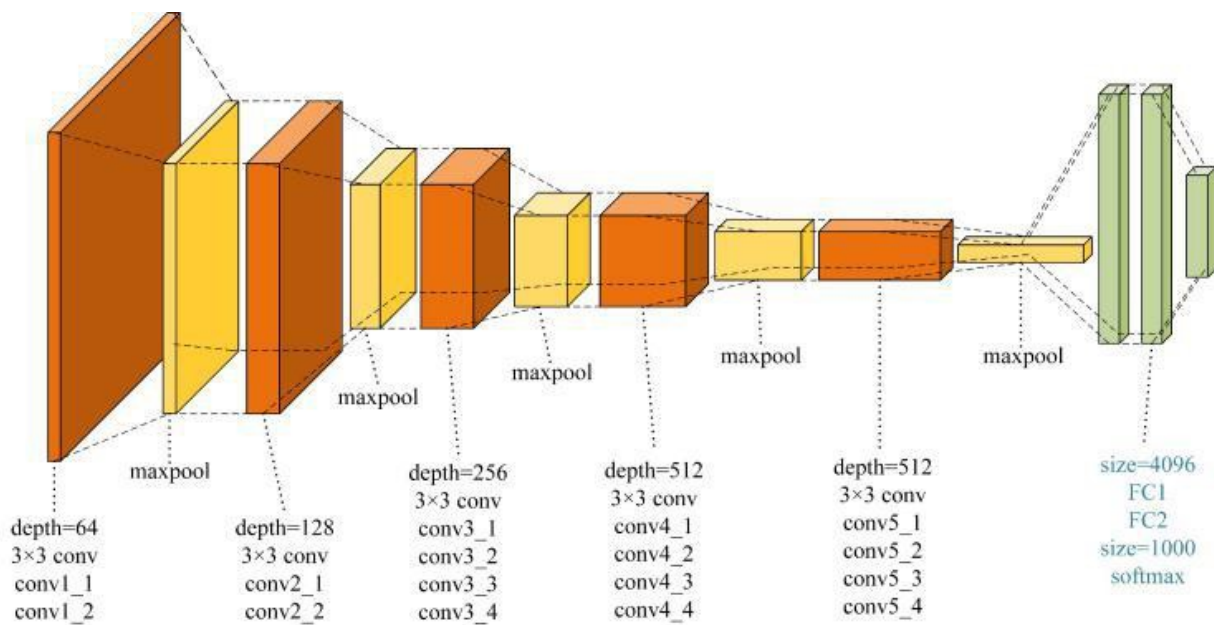


Abbildung 5: Aufbau Convolutional Neural Network VGG-19 (Yao, Li, & Qi, 2019)



(Gatys, Ecker, & Bethge, 2016) verwenden das in Abbildung 5 dargestellte VGG-19 Netzwerk, um den Stiltransfer durchzuführen. Sie haben das Netzwerk normalisiert, indem die Gewichte so skaliert wurden, dass die mittlere Aktivierung jedes Faltungsfilters über Bildern und Positionen gleich eins ist. Eine solche Neuskalierung kann für das VGG-Netzwerk durchgeführt werden, ohne dessen Ausgabe zu ändern, da es nur gleichgerichtete lineare Aktivierungsfunktionen und keine Normalisierung oder Zusammenfassung über Feature-Maps enthält. (Gatys, Ecker, & Bethge, 2016) haben keine Fully-connected Layer verwendet und den maximalen Pooling-Vorgang durch ein durchschnittliches Pooling ersetzt, da es etwas ansprechendere Ergebnisse liefert.

### 2.3.3 Beschreibung des Algorithmus

Abbildung 6 zeigt den von (Gatys, Ecker, & Bethge, 2016) entwickelten Style Transfer Algorithmus. Zuerst werden Inhalts- und Stilmerkmale extrahiert und gespeichert. Das Stilbild  $\vec{a}$  durchläuft das Netzwerk. Seine Stildarstellung  $A^l$  wird auf allen enthaltenen Ebenen berechnet und gespeichert (Abbildung 6, links). Das Inhaltsbild  $\vec{p}$  durchläuft ebenfalls das Netzwerk und die Inhaltsdarstellung  $P^l$  in einem Layer wird gespeichert (Abbildung 6, rechts). Dann durchläuft ein zufälliges weißes Rauschbild  $\vec{x}$  das Netzwerk und seine Stilmerkmale  $G^l$  und Inhaltsmerkmale  $F^l$  werden berechnet.  $G^l$  ist die Gram-Matrix, die durch Berechnung der Merkmalskorrelationen zwischen den Schichten definiert ist. Die Gram-Matrix enthält Stilinhalte wie Textur, Farbe usw. Die Definition lautet:

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l$$

Auf jeder Ebene  $l$ , die in der Stildarstellung enthalten ist, wird elementweise die mittlere quadratische Differenz zwischen  $G^l$  und  $A^l$  berechnet, um den totalen Stilverlust

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^L \omega_l E_l$$

zu erhalten (Abbildung 6, links).  $\omega_l$  sind Gewichtungsfaktoren jeder Schicht  $l$  zum Gesamtverlust. Der Gesamtverlust ist wie folgt definiert:

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - A_{ij}^l)^2$$

Auch die mittlere quadratische Differenz zwischen  $F^l$  und  $P^l$  wird berechnet, um den Inhaltsverlust

$$\mathcal{L}_{content}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} (F_{ij}^l - P_{ij}^l)^2$$

zu erhalten (Abbildung 6, rechts).  $i$  steht dabei für den  $i$ -ten Filter an Position  $j$  in Ebene  $l$ . Der Gesamtverlust

$$\mathcal{L}_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{content}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{style}(\vec{a}, \vec{x})$$

ist dann eine lineare Kombination zwischen dem Inhalt- und Stilverlust.  $\alpha$  und  $\beta$  sind Gewichtungsfaktoren für die Inhalts- bzw. Stilrekonstruktion. Seine Ableitung in Bezug auf die Pixelwerte kann unter Verwendung der Fehlerrückführung (Error Backpropagation) berechnet werden (Abbildung 6, Mitte). Dieser Gradient wird verwendet, um das Bild  $\vec{x}$  iterativ zu aktualisieren, bis es gleichzeitig

mit den Stilmerkmalen des Stilbilds  $\vec{a}$  und den Inhaltsmerkmalen des Inhaltsbilds  $\vec{p}$  (Abbildung 6, Mitte, unten) übereinstimmt. (Gatys, Ecker, & Bethge, 2016)

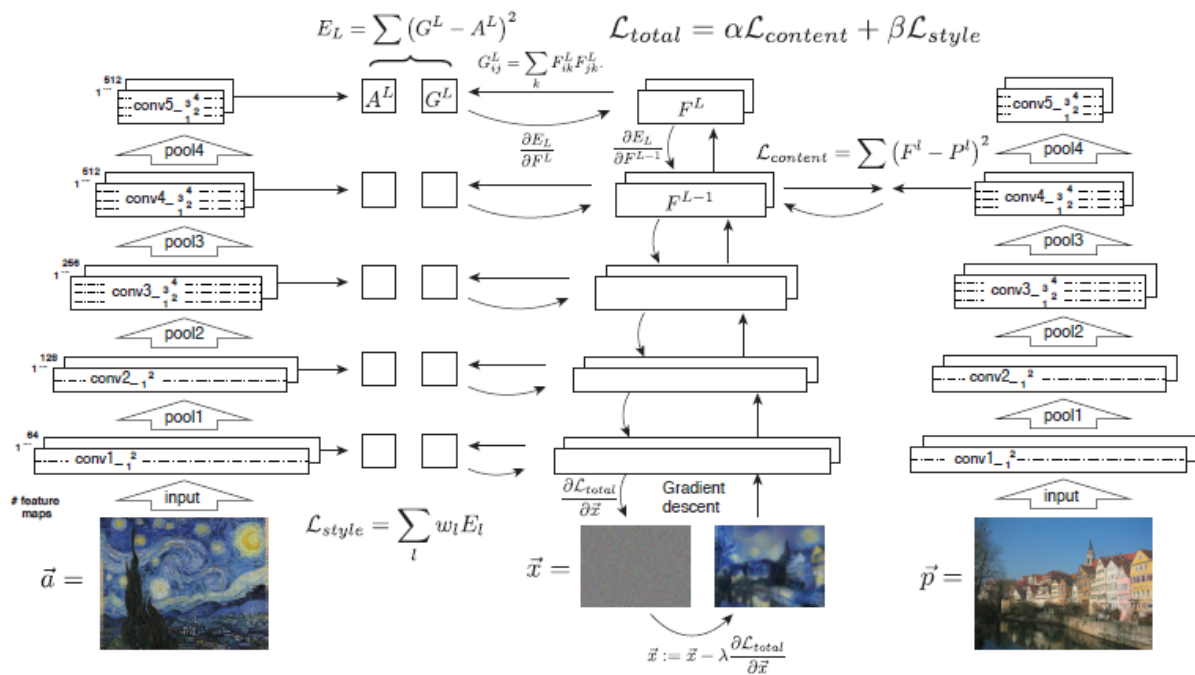


Abbildung 6: Style Transfer Algorithmus (Gatys, Ecker, & Bethge, 2016)

In Folgearbeiten haben (Gatys, Bethge, Hertzmann, & Shechtman, 2016) und (Gatys L. A., Ecker, Bethge, Hertzmann, & Shechtman, 2017) einen Algorithmus entwickelt, um die Farbe des Inhaltsbildes beizubehalten und verschiedene Stile entsprechend der semantischen Information auf verschiedene Regionen in Eingabebildern zu übertragen.

Die vorgestellte Methode von (Gatys, Ecker, & Bethge, 2016) ist ziemlich zeitintensiv, da mehrere bis tausende Iterationen benötigt werden, um jedes Mal ein neues Bild zu erzeugen. Um Zeit und Kosten zu sparen haben (Johnson, Alahi, & Fei-Fei, Perceptual Losses for Real-Time Style Transfer and Super-Resolution, 2016) einen Stilübertragungsalgorithmus entwickelt, der die Geschwindigkeit um drei Größenordnungen verbessern kann. Dieser wird als Fast Neural Style Transfer bezeichnet. Es werden zwei Netzwerke verwendet, ein Bildtransformationsnetzwerk und ein Verlustnetzwerk. Das Bildtransformationsnetzwerk setzt ein Pre-Training voraus, damit ein bestimmtes Inhaltsbild schnell einen festen Bildstil erzeugen kann. Das Verlustnetzwerk wird verwendet, um den Wahrnehmungsverlust zu berechnen. Der Wahrnehmungsverlust wird wiederum verwendet, um das Bildtransformationsnetzwerk zu trainieren und seine Parameter zu aktualisieren.

Einen ausführlichen Überblick zum Thema Neural Style Transfer und deren Verbesserungen und Erweiterungen des ursprünglichen Ansatzes bietet die Abhandlung von (Jing, et al., 2018). Sie stellen außerdem eine Taxonomie zur Einteilung der gegenwertig vorhandenen Algorithmen in diesem Bereich vor.

### 3 Methodik

Das Ziel der Arbeit besteht darin einen Arbeitsablauf zu entwickeln, der es ermöglicht, aus einer Bildaufnahme eines Objektes beliebigen Ursprungs mittels Bildverarbeitungsmethoden eine ikonographische Darstellung zu erzeugen. Um dieses Vorhaben umzusetzen sind drei Workflows entwickelt worden, die im vorliegenden Kapitel unter den Namen „Ölmalerei“, „Tuschezeichnung“ und „Silhouette“ konzeptionell näher vorgestellt werden. Zu Beginn dieses Abschnitts sind die dafür verwendeten technischen Komponenten beschrieben.

#### 3.1 Technische Komponenten

Für die praktische Umsetzung des Projektes wurde sich für das rasterbasierte Bildbearbeitungsprogramm GIMP in der Version 2.10.12 entschieden. Die Abkürzung steht für GNU Image Manipulation Program<sup>4</sup>. Neben der weit verbreiteten kommerziellen, von Adobe entwickelten und vertriebenen Software Photoshop<sup>5</sup> stellt GIMP eine gute und vor allem freie Open Source Alternative dar. Aufgrund einer regen Community gibt es eine Vielzahl frei zugänglicher Plug-Ins und Skripte, um den internen Programmumfang zusätzlich zu erweitern. (Hartl, 2015) hat sich in seiner Arbeit ausführlich mit GIMP beschäftigt und zeigt dessen Möglichkeiten zur Grafikerzeugung und Bildmanipulation sehr gut auf.

Im Rahmen dieser Arbeit wurde im Speziellen auf eine Filtersammlung namens G'MIC zurückgegriffen. Das Kurzwort steht für GREYC's Magic for Image Computing. Es handelt sich hierbei, um ein voll funktionsfähiges Open Source-Framework für die Bildverarbeitung, welches unter der CeCILL-Lizenz für freie Software vertrieben wird. Das Programm bietet zudem mehrere Benutzeroberflächen, unter anderem auch für GIMP (siehe Abbildung 7) zum Konvertieren, Manipulieren, Filtern und Visualisieren generischer Bilddatensätze an.<sup>6</sup> Außerdem existiert ein sog. Command-Line Interface (CLI) mit dem es möglich ist Arbeitsprozesse zu automatisieren. Speziell dieser Punkt ist für die Arbeit sehr interessant und wird im Kapitel 4.2 näher untersucht und umgesetzt.

Abschließend ist angedacht, die Ergebnisse als Symbole auf einer Karte platziert zu präsentieren. Hierfür wurde als Datengrundlage eine vereinfachte OSM-Karte verwendet und in QGIS, dem freien Open Source Geographischen Informationssystem<sup>7</sup> entsprechend aufbereitet. Die nun folgend beschriebenen Workflows sollen dabei drei Maßstabsbereiche definieren.

#### 3.2 Kriterien der Bildauswahl

Bei der Auswahl der geeigneten Bilder ist auf eine einheitliche Ausleuchtung der Szenerie zu achten, damit möglichst keine farblichen Divergenzen zwischen den Ergebnissen auftauchen. Als Aufnahmeperspektive ist die Frontansicht zu wählen, da diese sich für ikonografische Zeichen am besten eignet. Alternativ ist auch die Schrägansicht bei komplexeren Objekten interessant. Es ist außerdem auf eine einheitliche Bildgröße für den Workflow-Einsatz zu achten.

---

<sup>4</sup> <https://www.gimp.org/>

<sup>5</sup> <https://www.adobe.com/de/products/photoshop.html>

<sup>6</sup> <https://gmics.eu/>

<sup>7</sup> <https://www.qgis.org/de/site/>

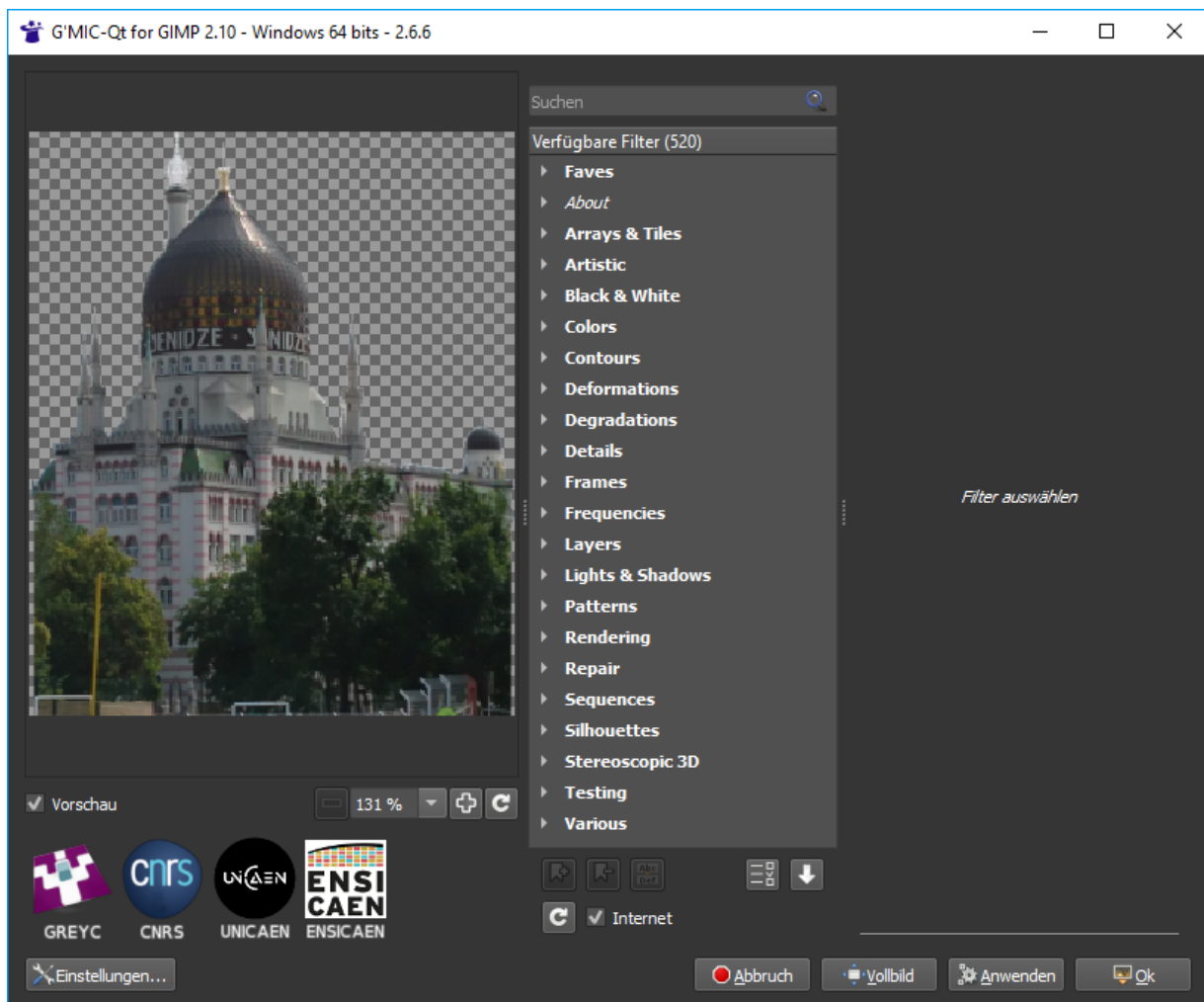


Abbildung 7: User Interface G'MIC für GIMP

### 3.3 Workflow „Ölmalerei“

Die Herausforderung besteht darin einen möglichst hohen Level der Abstraktion vom Originalfoto und damit eine quasi Hand gezeichnete Illustration zu erreichen. Die Abbildung 8 zeigt die daraus entwickelten generischen Hauptschritte, die dann im praktischen Teil der Arbeit durch den Einsatz entsprechender Bildbearbeitungsfilter umgesetzt werden sollen.

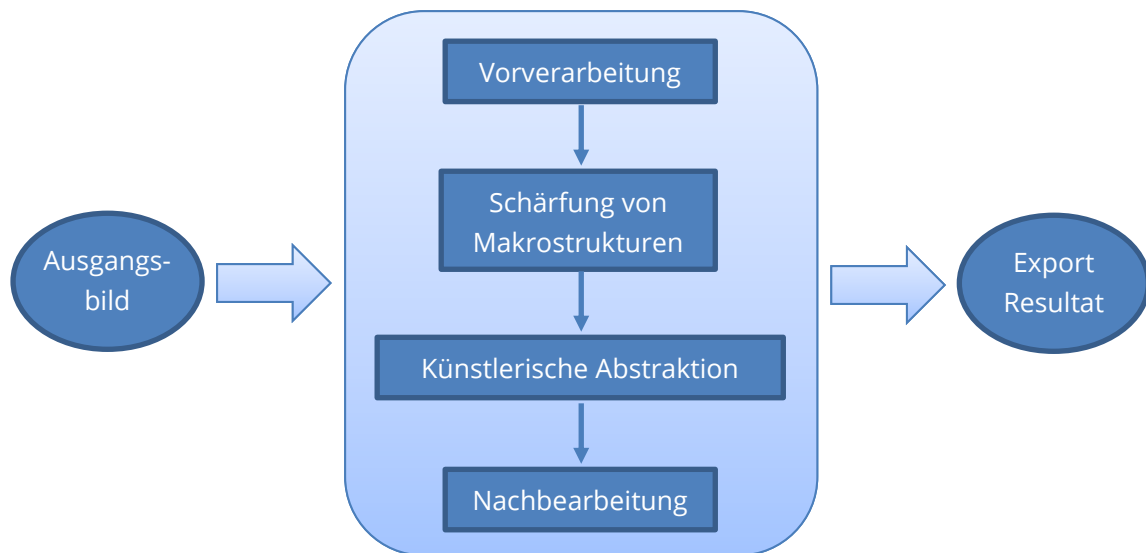


Abbildung 8: Arbeitsschritte Workflow "Ölmalerei"

#### **Vorverarbeitung**

In einem ersten Arbeitsschritt soll das Objekt von Interesse für die Weiterverarbeitung vorbereitet werden. Hierzu wird per Auswahlwerkzeug das Objekt gegenüber dem Hintergrund freigestellt und als neues Bild exportiert. Die einzelnen Ausgangsbilder können dabei qualitativ sehr unterschiedlich sein. Damit die Ergebnisse letztlich möglichst identisch ausfallen ist zunächst eine einheitliche Skalierung angedacht. Außerdem wird mit einer Helligkeits- und Farbanpassung auf Unterschiede in einzelnen Bildbereichen reagiert, um so eine Angleichung zu erzielen.

#### **Schärfung von Makrostrukturen**

Wie es der Titel bereits initiiert soll im weiteren Verarbeitungsprozess eine Betonung der Makrostrukturen in einem Bild im Vordergrund stehen. Meist ist es so, dass die vorverarbeitenden Schritte zu einem verwaschenen milchigen Gesamteindruck führen. Für die weiteren Schritte ist es aber notwendig deutlich hervortretende Konturen zu haben. Aus diesem Grund kommen an dieser Stelle schärfende Filter zum Einsatz, um die markanten Strukturen eines Objektes hervorzuheben und zu verbessern.

#### **Künstlerische Abstraktion**

Dieser Bereich stellt das Kernstück der Bearbeitung dar. Hier wird durch den Einsatz künstlerischer Filter und einem entsprechenden Ebenenmanagement versucht den malerischen Gesamteindruck herzustellen. Während des Entwicklungsprozesses haben sich vor allem zwei Herangehensweisen und deren Filter herauskristallisiert. Zum einen das Posterisieren (Posterize) und zum anderen das

Gravieren (Engrave). Die Posterisation bezeichnet dabei den Effekt einer Tonwerttrennung bei gleichzeitiger Reduzierung der Tonwerte (Wolf, 2012, S. 293). Nachdem die Anzahl der Farben reduziert, das Erscheinungsbild somit abstrahiert ist, sollen mit einem Gravur-Filter wichtige Kanten des Objektes extrahiert werden. Durch den Einsatz eines Sketch-Filters soll zudem der zeichnerische Eindruck verstärkt werden. Dem bereits erwähnten Ebenenmanagement fällt dabei im Zusammenspiel der einzelnen Ergebnisse eine sehr wichtige Rolle zu. Im praktischen Teil dieser Arbeit wird darauf explizit eingegangen.

### **Nachbearbeitung**

Im Nachgang sollen nun noch einige Verschönerungsarbeiten stattfinden. Unter den Schlagworten Verwischen und Glätten können die folgenden Arbeitsprozesse sehr gut zusammengefasst werden. Mithilfe eines Blur-Filters und anschließender Faserextraktion soll ein glänzendes Erscheinungsbild generiert werden. Das abschließende Verwenden eines Smooth-Operators erzeugt ein geglättetes weichgezeichnetes Gesamtergebnis. Auch hier spielt die Organisation der Ebenen eine entscheidende Rolle. Es sei ebenfalls auf Kapitel 4 den konkreten Anwendungsfall verwiesen.

Die Resultate sollen schlussendlich in einer Beispielkarte platziert dargestellt werden. Hierfür gibt es Überlegungen der Notwendigkeit, die Objekte mit einer Umrisslinie zu versehen, um eine Freistellung gegenüber der Umgebung zu erzielen. Ebenso kann Schatten einen 3D-Eindruck und damit eine Abhebung vom Hintergrund generieren.

Zu guter Letzt erfolgt der Export in ein beliebiges Bildformat. In der Arbeit wurde sich für das PNG-Format entschieden, da es im Gegensatz zum häufig verwendeten JPG-Format eine verlustfreie Datenkompression ermöglicht.

## **3.4 Workflow „Tuschezeichnung“**

Die bereits in Kapitel 3.3 im Detail beschriebenen und in Abbildung 8 definierten Arbeitsschritte können ohne Weiteres auch auf die Darstellung einer Tuschezeichnung übertragen werden. Lediglich der Block der „Künstlerischen Abstraktion“ unterscheidet sich an dieser Stelle. Im Ergebnis soll eine Art skizzenhafte Pinselzeichnung generiert werden. Hierfür wird nach der Vorverarbeitung und Detailextraktion die Entsättigung der Farben erfolgen. Mittels eines morphologischen Filters ist es angedacht sehr dunkle Bereiche im Bild zu entfernen und die Darstellung schließlich durch einen Stempel-Effekt in eine schwarz-weiße Tuschezeichnung zu überführen. Im Nachgang sollen die Schritte zur Nachbearbeitung ebenfalls erfolgen.

## **3.5 Workflow „Silhouette“**

Der Schattenriss stellt im Grunde die einfachste der drei Entwicklungsstufen dar. Konzeptionell ist es lediglich notwendig das Objekt vollständig auszuwählen und diese Auswahl komplett schwarz einzufärben. Bei der praktischen Umsetzung wurde dafür ein Filter verwendet, der es ermöglicht grundlegende Anpassungen die Farbe betreffend vorzunehmen. Auch hier sind im Nachhinein eine Umrisslinie und ein Schattenwurf um das Objekt herum angedacht.

## 4 Praktischer Teil

In diesem Kapitel erfolgt die konkrete Umsetzung der vorgestellten Workflows in GIMP. Dabei werden anhand diverser Beispiele jene Einzelschritte und deren Resultate aufgezeigt. Abgerundet wird die praktische Arbeit durch den Versuch der Automatisierung und einer Kartendarstellung.

### 4.1 Konkrete Umsetzung

Die in Kapitel 3 dargestellte Methodik wird an dieser Stelle in aller Ausführlichkeit technisch umgesetzt. Dabei werden zu jedem Workflow die Herangehensweise und der entsprechende Einsatz von Filtern, sowie deren Kombinationen beschrieben. Es wurden Filter aus der Filtersammlung G'MIC für GIMP verwendet. Hierzu wurde das entsprechende Plug-In von der Homepage<sup>8</sup> heruntergeladen und installiert. Abrufbar ist es schließlich in der GIMP Benutzeroberfläche über Menü „Filter“ und „G'MIC-Qt...“.

Zur Demonstration werden folgend die in Abbildung 9 dargestellte Dresdner Frauenkirche und in Abbildung 10 dargestellte Katholische Hofkirche in Dresden als Ausgangsbilder verwendet. Es handelt sich um Quellbilder aus dem Internet. Explizit wurden diese beiden Fotos ausgewählt, um auf die qualitativen Unterschiede in den Ausgangsbildern, hier die erkennbaren hell-dunkel Differenzen im Rahmen der Workflowentwicklung zu berücksichtigen und zu reagieren.

Das Ziel am Ende besteht darin, möglichst identisch aussehende Resultate zu erhalten, egal ob eine Hochglanzaufnahme per Spiegelreflexkamera oder die eigene, etwas verschwommene Aufnahme mit dem Smartphone die Grundlage bildet.



Abbildung 9: Ausgangsbild Frauenkirche Dresden (Aufnahme: Stadtrundfahrt.com<sup>9</sup>)

<sup>8</sup> <https://gmic.eu/download.shtml>

<sup>9</sup> Kuppelaufstieg Frauenkirche Dresden GmbH  
(<https://www.stadtrundfahrt.com/dresden/kuppelaufstieg-frauenkirche/>)





Abbildung 10: Katholische Hofkirche Dresden (Aufnahme: Heinz-Josef Lücking, Lizenz: CC BY-SA 3.0 de)

#### 4.1.1 Workflow „Ölmalerei“

##### **Vorverarbeitung**

Nachdem das geeignete Foto ausgewählt wurde, steht der praktischen Arbeit nichts mehr im Wege. Wie in Kapitel 3.3 beschrieben erfolgt zunächst eine Freistellung des gewünschten Objektes. Hierfür stellt GIMP mehrere Werkzeuge zur Verfügung (Abbildung 11). Neben einer rechteckigen und elliptischen Auswahl, die für diesen Anwendungsfall jedoch ungeeignet sind, gibt es zum einen die „Magnetische Schere“ und zum anderen die „Freie Auswahl“. Die „Magnetische Schere“, auch als Intelligente Schere bezeichnet, versucht beim Setzen der Kontrollpunkte die optimale Kante und Kurve vom vorherigen zum aktuellen Stützpunkt zu finden (Wolf, 2012). Dass das nicht immer reibungslos funktioniert, zeigt die Abbildung 12. Aus diesem Grund wurde sich für die „Freie Auswahl“, auch Lasso-Werkzeug entschieden. Bereiche eines Bildes können so ebenfalls durch das Setzen von Stützpunkten frei nach Bedarf ausgewählt werden. Ist die Markierung abgeschlossen, wird diese über Menü „Bearbeiten“ und „Einfügen als“, als ein „Neues Bild“ in GIMP eingefügt.



Abbildung 11: Auswahlwerkzeuge in Gimp (von links: Rechteckige Auswahl, Elliptische Auswahl, Freie Auswahl, Zauberstab, Nach Farbe auswählen und Magnetische Schere)



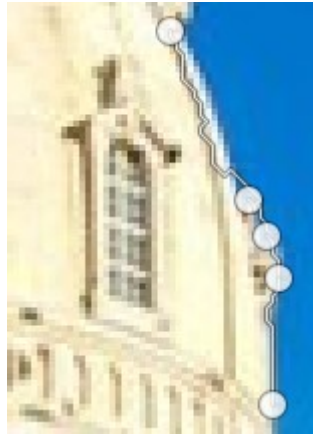


Abbildung 12: Auswahl mit "Magnetischer Schere"

Anschließend erfolgt über Menü „Bild“ und „Bild skalieren...“ die Anpassung der Größe der Bilder. Die meisten verwendeten Filter sind größenabhängig. Das heißt, die Resultate unterscheiden sich je nach Format. Untersuchungen im Laufe der Entwicklung haben gezeigt, dass eine maximale Höhe von 300px anzustreben ist. Aufnahmen, die in ihrer Höhe größer sind, als 300px müssen entsprechend angepasst werden. Bei Bildern unterhalb dieser Höhe ist dies nicht notwendig.

Das Ergebnis der bisherigen Bearbeitung zeigt Abbildung 13 für die Untersuchungsobjekte.



Abbildung 13: Ausgangsbilder nach Freistellung und Skalierung

Im nächsten Schritt folgt eine Helligkeitsanpassung. Dunkle Bereiche sollen so minimiert und insgesamt eine Angleichung erzielt werden. In G'MIC steht dafür unter der Rubrik „Colors“ der Filter „Normalize Brightness“ bereit. Für die Frauenkirche und Hofkirche wurden die in Tabelle 1 dargestellten Parameter verwendet. Anschließend wurde der „Retinex“-Filter aus der selben

Kategorie eingesetzt, um noch einmal das gesamte Erscheinungsbild aufzuhellen. Das Ergebnis im Vergleich zeigt die Abbildung 14. Zusätzlich wurde mit „Color Temperature“ (Stufe: 10) dem Foto eine warme und extrem rötliche Färbung verliehen.

	Frauenkirche	Hofkirche
<b>Bright</b>	2	2
<b>Area</b>	10	100
<b>Smooth</b>	0	0
<b>Channels</b>	YCbCr	HSV
<b>Mask</b>	Lighten	Lighten

Tabelle 1: Parameter Filter Normalize Brightness

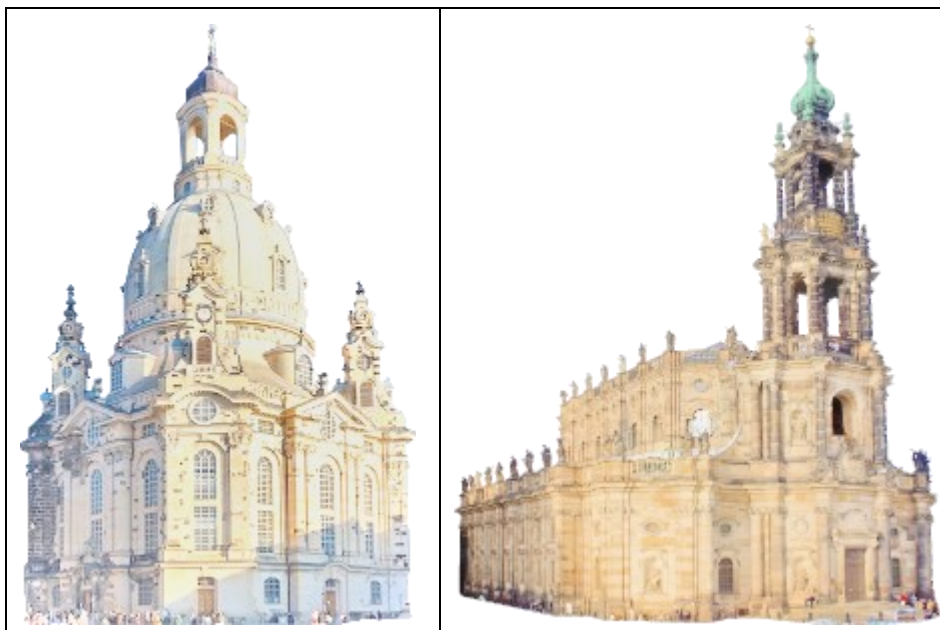


Abbildung 14: Ergebnis nach Einsatz Normalize Brightness und Retinex

### Schärfung von Makrostrukturen

Wie bereits angesprochen führen die vorbereitenden Schritte zu einem verschwommenen unscharfen Eindruck. Um dem entgegenzuwirken, werden im Folgenden nacheinander aus der Kategorie „Details“ die Filter „Equalize Local Histograms“, „Freaky Details“ und abschließend „Sharpen [Deblur]“ eingesetzt. Es wurden jeweils die Standardparameter unverändert beibehalten und sind deshalb nicht explizit mit angegeben. Das Ergebnis dieses Vorgangs zeigt Abbildung 15.

### Künstlerische Abstraktion

Im malerischen Teil der Bearbeitung angekommen, wird mit dem „Posterize“-Filter aus der Kategorie „Artistic“ die Anzahl der Farben auf 12 reduziert. Außerdem erzeugt der Filter einen ersten zeichnerischen Eindruck (Abbildung 16), da er die Kanten reduziert (30%) und durch einen Smoothness-Faktor (150) das Bild insgesamt weichzeichnet und damit fließende Übergänge entstehen. Auch hier wurden die Default-Einstellungen verwendet.

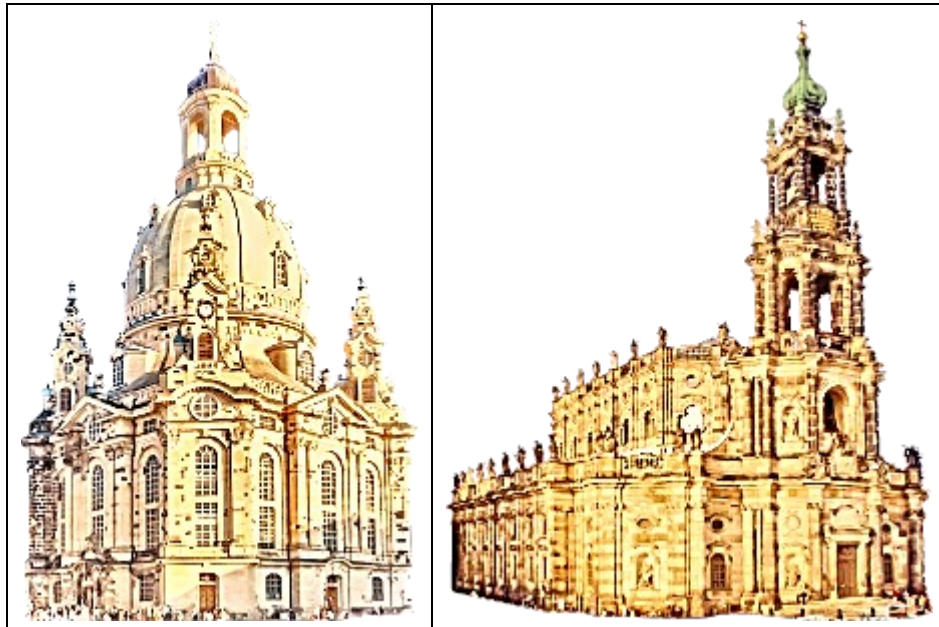


Abbildung 15: Ergebnis nach Schärfung von Makrostrukturen



Abbildung 16: Ergebnis Filter Posterize

Jedes Bild, das in GIMP hineingeladen und bearbeitet wird, besitzt mindestens eine Ebene. Die sog. Hintergrundebene wird im Ebenen-Dialog angezeigt. Auf dieser liegt das Bild. Hier gibt es die Möglichkeit weitere Ebenen hinzuzufügen (Wolf, 2012, S. 353). Diese Tatsache ausnutzend wird die aktuelle Ebene mit dem Posterize-Ergebnis einmal per „Ebene duplizieren“ kopiert (Abbildung 17). Auf die erstellte Kopie wird aus der Rubrik „Black & White“ der Filter „Engrave“ angewendet. Die Einstellungen der Parameter zeigt Tabelle 2. Der Lightness-Wert steuert die Stärke der Farbgebung und Helligkeit. Im Ergebnis entsteht neben der schwarz-weißen Gravur eine zusätzliche Ebene mit den

Farbinformationen (Color background). Beide Ebenen werden durch den Ebenenmodus „Nur Abdunkeln“ überblendet. Bei diesem Modus werden die Pixel der oberen und unteren Ebene miteinander verglichen. Verwendet wird immer der kleinere Wert. Schwarze Pixel ergeben schwarze Bereiche, weiße Pixel ändern nichts am Ergebnis (Wolf, 2012, S. 449). Um weiterarbeiten zu können ist es notwendig diese zwei Ebenen per „Nach unten vereinen“ zusammenzufügen. Anschließend wird der Ebenenmodus auf „Multiplikation“ gesetzt. Bei diesem Modus werden die Farbwerte der oberen und unteren Ebene multipliziert und durch 255 dividiert (Wolf, 2012, S. 442). Die vereinte Ebene und die übrig gebliebene Posterize-Ebene ergeben das in Abbildung 18 dargestellte Bild.

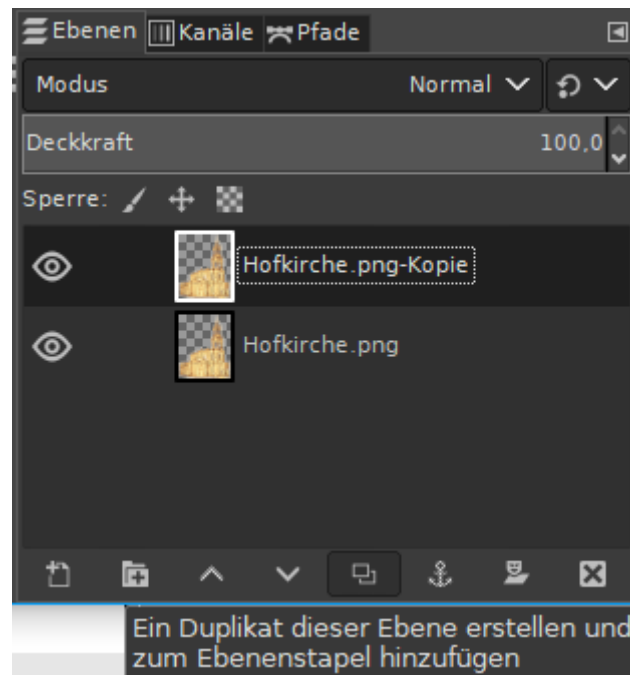


Abbildung 17: Ebenen-Dialog in GIMP

	Frauenkirche	Hofkirche
<b>Black &amp; White foreground:</b>		
Radius	1	1
Density	15	15
Edges	2	2
Coherence	2	2
Threshold (%)	70	70
Minimal Area	0	0
Flat Regions Removal	0	0
<b>Color background:</b>	Add Color Background	
Quantization	5	5
Shading	1	1
Hue	0	0
Saturation (%)	15	15
Lightness (%)	-30	20
Anti-Aliasing	x1.5	x1.5

Tabelle 2: Parameter Filter Engrave



Abbildung 18: Ergebnis nach Engrave und Anpassung Ebenenmodus

Mit Rechtsklick auf die obere Ebene im Ebenen-Dialog und „Neu aus Sichtbarem“ wird eine neue Ebene aus dem, was aktuell zu sehen ist, erstellt. Das erleichtert das Weiterarbeiten ohne vorherige Schritte zu verändern. Mit dem Filter „Sketch“ aus der G'MIC-Kategorie „Artistic“ wird zum Abschluss der künstlerischen Abstraktion eine Art Buntstiftzeichnung generiert (Abbildung 19). Die Grundeinstellungen wurden nicht verändert.

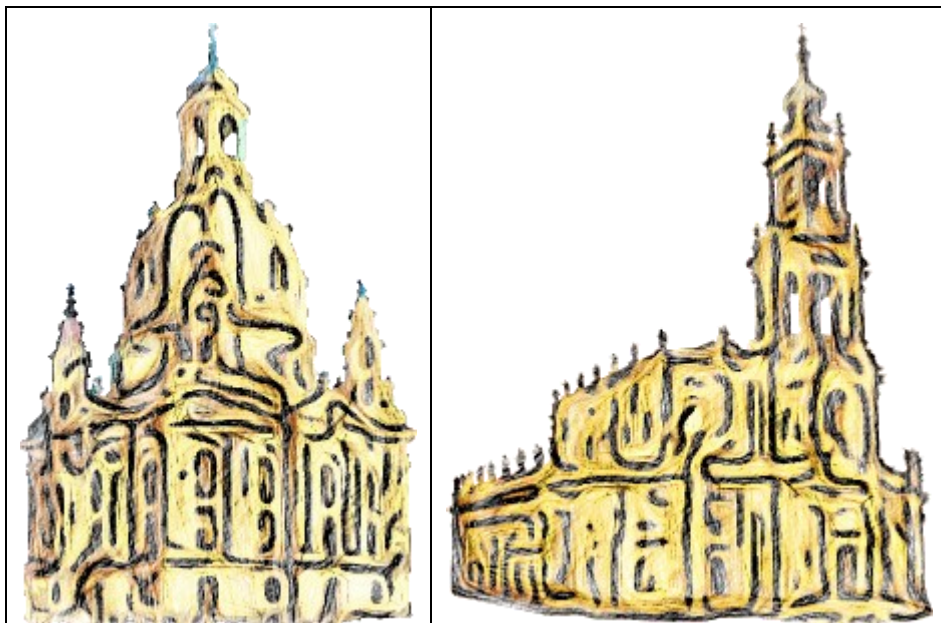


Abbildung 19: Ergebnis Filter Sketch



## Nachbearbeitung

Die letzten Arbeitsschritte dienen der Abrundung des Erscheinungsbildes. Hierfür wird zunächst das Sketch-Ergebnis zweimal dupliziert. Durch Anwendung von „Blur [Bloom]“ aus der Klasse „Degradations“ auf die oberste Ebene entsteht ein verwaschenes nebeliges Ergebnis. Anschließend wird der Ebenenmodus auf „Faser extrahieren“ umgestellt. Dieser Modus entfernt die Körnigkeit und legt diese Struktur in einer Ebene zur Weiterverarbeitung ab. Außerdem ist es möglich, Bildern ein plastisches Aussehen zu verleihen (Wolf, 2012, S. 446). Genau das, ist das Ziel der folgenden Bearbeitung. Die obere Ebene im Anschluss „Nach unten vereinen“. Mit G’MICs „Black & White“-Filter wird die Farbe entsättigt (Abbildung 20) und anschließend der Ebenenmodus auf „Nachbelichten“ gesetzt. Neue Ebene aus Sichtbarem erstellen und mit Smooth [Thin Brush] aus der Kategorie „Repair“ weichzeichnen. So wird die letzte Körnung aus dem Ergebnis entfernt. Abbildung 21 zeigt das Endresultat des Workflows „Ölmalerei“.

Für die beabsichtigte Darstellung auf einer Karte wurde überlegt, der Signatur eine Umrisslinie und einen Schatten zur besseren Abhebung gegenüber den umliegenden Objekten hinzuzufügen. Für dieses Vorhaben stellt G’MIC die Filter „Stroke“ (Rubrik: „Layers“) und „Drop Shadow“ (Rubrik: „Lights & Shadows“) zur Verfügung. Erst genannter Filter erstellt die Kontur. Als Stärke (Thickness) wurden 12px festgelegt. Die restlichen Parameter bleiben unverändert. Dem Ergebnis wird ein Layer mit der Umrisslinie zugefügt. Für die Weiterverarbeitung ist es auch an dieser Stelle notwendig die Ebene nach unten zu vereinen oder eine neue aus Sichtbarem zu erzeugen. Im Anschluss wird per „Drop Shadow“ (Default-Einstellungen) der Ansicht ein Schatten zugefügt (Abbildung 22).

Die Erstellung ist mit diesem Schritt endgültig abgeschlossen und kann in GIMP über Menü „Datei“ und „Exportieren als...“ in eine Vielzahl an Bildformaten exportiert werden. Im Rahmen dieser Arbeit ist das PNG-Rastergrafikformat verwendet worden.

Die Abbildung 23 zeigt abschließend noch einen Vergleich zwischen Originalaufnahme und der entstandenen „Ölmalerei“ in verkleinerter Form. Mehr zur Kartenanwendung in Kapitel 4.3.

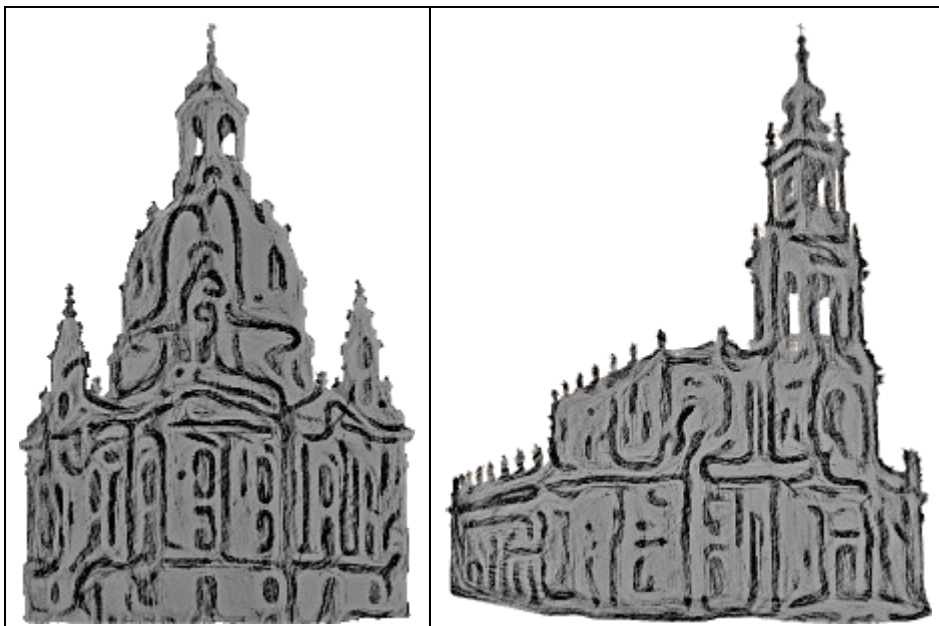


Abbildung 20: Extrahierte Faser, Farben entsättigt



Abbildung 21: Endergebnis Workflow Ölmalerei



Abbildung 22: Endergebnis mit Konturlinie und Schatten

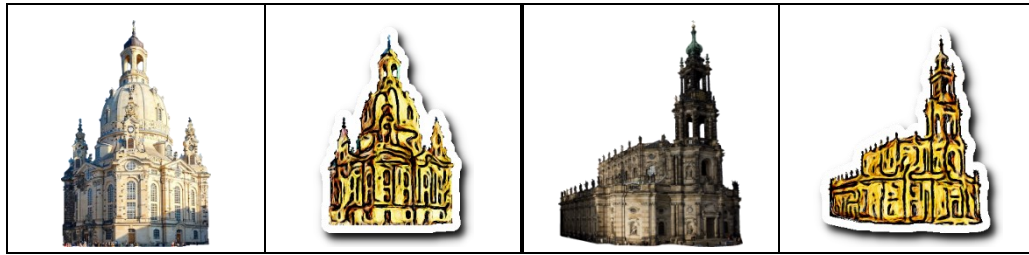


Abbildung 23: Vergleich Ausgangsbild und Ergebnis Ölmalerei (verkleinert)

#### 4.1.2 Workflow „Tuschezeichnung“

Die Vorverarbeitung und Detailherausarbeitung des Ölmalerei-Workflows können ohne Einschränkungen übernommen werden. Das heißt, die Schritte sind identisch. Fortgesetzt wird an dieser Stelle der Bearbeitung nach Filtereinsatz „Sharpen [Deblur]“ mit dem künstlerischen Block. Allerdings ist dieser weniger umfangreich.

Beginnend mit dem Filter „Black & White“ werden die Farben entsättigt (Abbildung 24). Anschließend erfolgt mit dem „Morphological Filter“ aus der G'MIC-Kategorie „Contours“ das Herauslösen dunkler Strukturen im Bild (Abbildung 25). Für den Parameter „Action“ wurde „Dilation - Original“ gewählt. Außerdem wurde die Stärke, Parameter „Size“ für die Frauenkirche auf 3 und für die Hofkirche auf 5, sowie der Haken bei „Negative“ gesetzt. Der Rest bleibt unverändert.

Anschließend wird mit dem „Stamp“-Filter (Kategorie: „Black & White“) eine Art schwarz-weißer Stempel-Effekt generiert (Abbildung 26). Mit Threshold kann dabei die Ausprägung der Konturen gesteuert werden. In beiden Fällen wurden die Standardwerte (Auto-Threshold bei 70) beibehalten.

In der Nachbearbeitung wird lediglich ein Glättungsfilter aus der Rubrik „Repair“ namens „Smooth [Antialias]“ angewendet, um den Gesamteindruck noch ein wenig weich zu zeichnen. Das Endergebnis des Workflows „Tuschezeichnung“ zeigt Abbildung 27.

Für die Kartendarstellung folgt schließlich das Hinzufügen der Konturlinie mit „Stroke“ und der Schatten mit „Drop Shadow“. Die genaue Beschreibung hierzu im Kapitel 3.1.1 unter Nachbearbeitung. Abbildung 28 zeigt das Resultat im Vergleich zum Ausgangsbild und in verkleinerter Ansicht.

#### 4.1.3 Workflow „Silhouette“

Der Workflow umfasst einen Bearbeitungsschritt. In GIMP das Bild öffnen. Über das Menü „Farben“ und „Einfärben“, die Farbe Schwarz auswählen und komplett einfärben.

Im Hinblick auf den angestrebten Automatisierungsprozess (siehe Kapitel 4.2) wurde auch hier ein Filter aus G'MIC verwendet. Mit „Basic Adjustments“ unter „Colors“ wird einfach, aber wirkungsvoll der Helligkeits-(Brightness) und Kontrast-Wert komplett auf -100% heruntergeregelt.

Im Anschluss daran Konturlinie und Schatten, wie bereits dargestellt, für die Kartenrepräsentation hinzufügen. Das Ergebnis zeigt Abbildung 29 im Vergleich zum Ausgangsfoto.



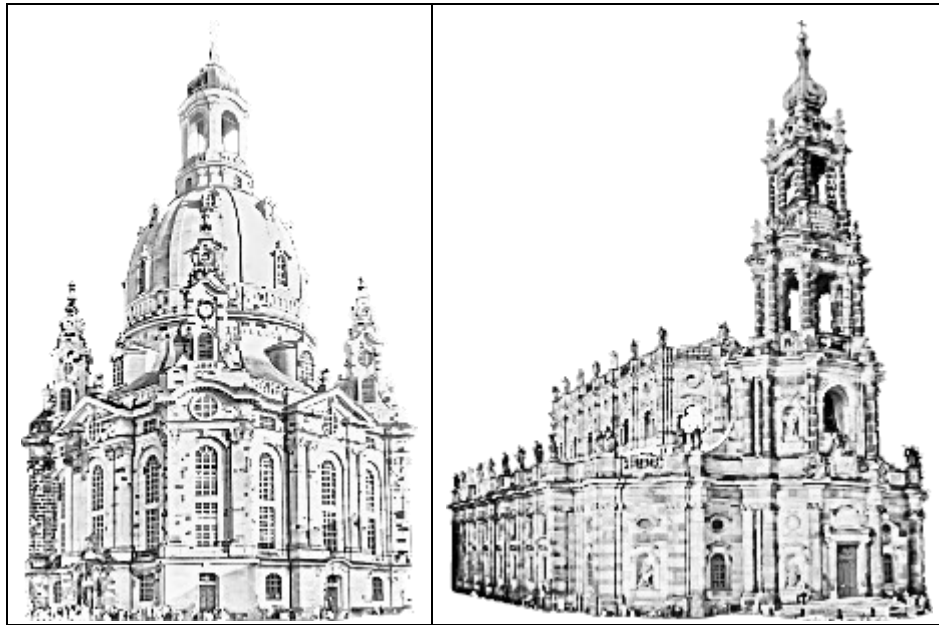


Abbildung 24: Entsättigung der Farben

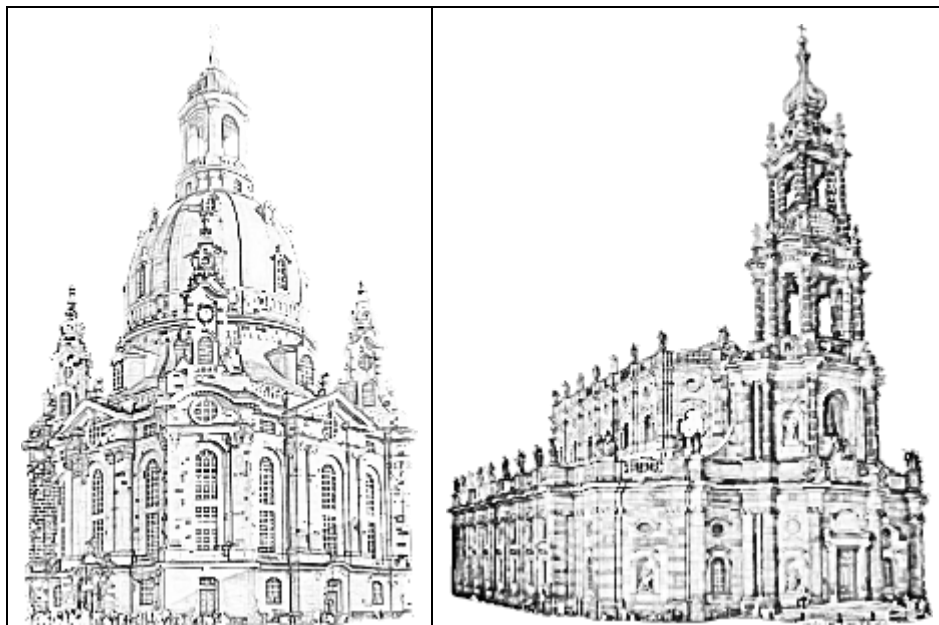


Abbildung 25: Ergebnis Morphological Filter

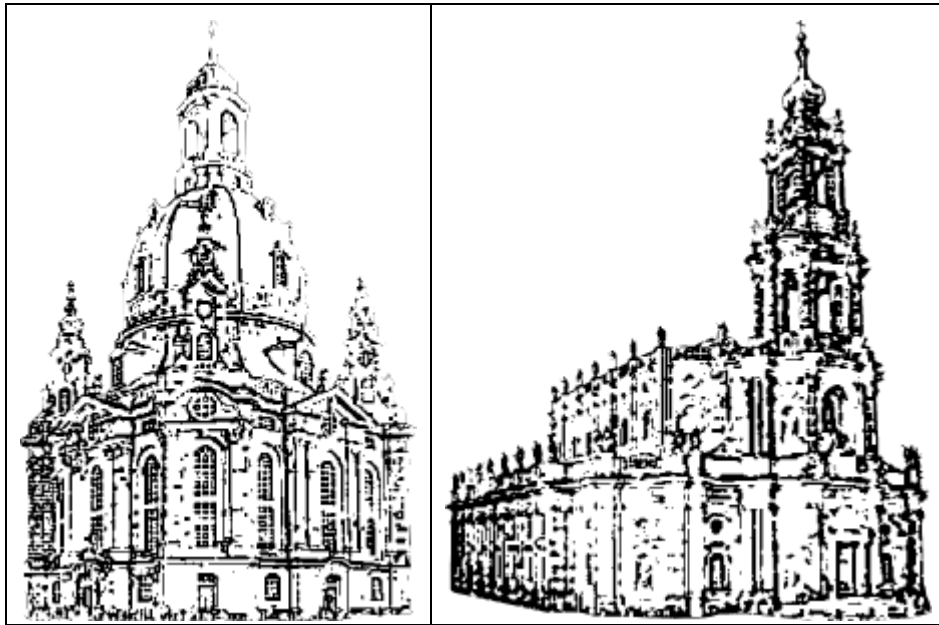


Abbildung 26: Ergebnis Filter Stamp



Abbildung 27: Endergebnis Workflow Tuschezeichnung

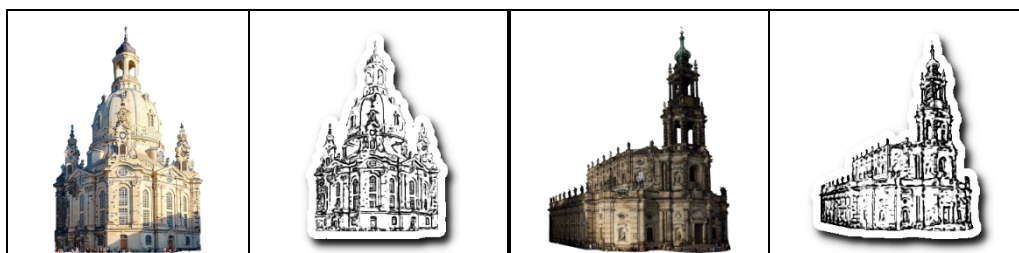


Abbildung 28: Vergleich Ausgangsbild, Endergebnis Tuschezeichnung mit Konturlinie u. Schatten (verkleinert)

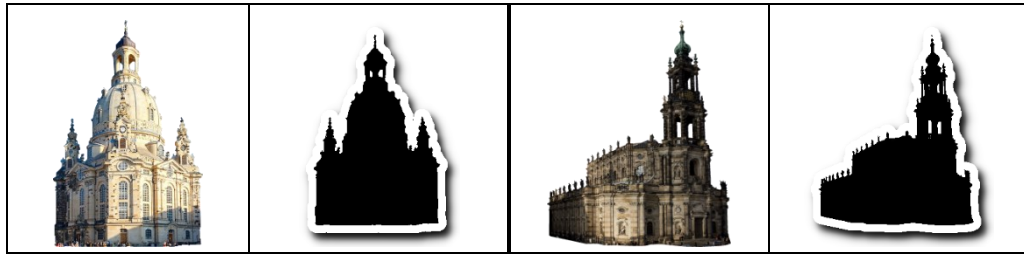


Abbildung 29: Vergleich Ausgangsbild und Endergebnis Silhouette mit Konturlinie und Schatten (verkleinert)

## 4.2 Implementierung eines Automatisierungsprozesses

Im vorliegenden Abschnitt wird ein Ansatz zur Automatisierung der Workflows beschrieben. Es wurde bereits darauf hingewiesen, dass für die Überführung der Originalaufnahmen in die jeweilige abstrahierte Darstellung G'MIC-Filter verwendet wurden. Tabelle 3 zeigt für jeden Filter bzw. Arbeitsschritt den Aufwand und definiert damit einen möglichen Grad der Automatisierung.

Filter/Arbeitsschritt	Bearbeitungsaufwand	Automatisierbarkeit
<b>Objektfreistellung</b>	sehr hoch	nein
<b>Skalierung</b>	gering	nein
<b>Normalize Brightness</b>	mittel	ja, mit Standardwerten
<b>Retinex</b>	gering	ja
<b>Color Temperature</b>	gering	Ja
<b>Equalize Local Histograms</b>	gering	Ja
<b>Freaky Details</b>	gering	Ja
<b>Sharpen [Deblur]</b>	gering	Ja
<b>Posterize</b>	gering	ja
<b>Engrave</b>	mittel	ja, mit Standardwerten
<b>Sketch</b>	gering	Ja
<b>Blur [Bloom]</b>	gering	ja
<b>Black &amp; White</b>	gering	Ja
<b>Smooth [Thin Brush]</b>	gering	ja
<b>Morphological Filter</b>	mittel	ja, mit Standardwerten
<b>Stamp</b>	mittel	ja, mit Standardwerten
<b>Smooth [Antialias]</b>	gering	ja
<b>Basic Adjustments</b>	gering	ja
<b>Stroke</b>	gering	ja
<b>Drop Shadow</b>	gering	ja

Tabelle 3: Kriterien der Reproduzierbarkeit

Fazit dieser Auswertung ist, dass eine Teilautomatisierung nach der Objektfreistellung und Skalierung erfolgen kann, indem allen Filtern repräsentable Standardparameter zugewiesen bekommen.

Über das Command-Line Interface (CLI)<sup>10</sup> von G'MIC wird im Folgenden der Prozess realisiert. Hierzu muss das heruntergeladene zip-Archiv (*gmic\_cli\_win64.zip*) in einen Ordner auf Laufwerk C:\ entpackt werden. Beispiel:

```
C:\Users\Ron\gmic-cli-2.6.7-win64
```

Anschließend wird G'MIC über die Kommandozeile von Windows angesprochen. Dafür die Eingabeaufforderung öffnen und mit dem Befehl

```
cd C:\Users\Ron\gmic-cli-2.6.7-win64
```

das Verzeichnis ansprechen. Jetzt können Befehle eingegeben und mit Enter ausgeführt werden.

Beispielanweisung:

```
gmic.exe -input D:\Bilder\Test.png -my_command -output D:\Bilder\Ergebnis.png
```

Mit *gmic.exe* wird das Programm ausgeführt und durch den input-Befehl ein Bild aus dem angegebenen Verzeichnis geladen. Auf dieses Bild wird die Funktion „my\_command“ angewendet. Das Resultat erscheint dann per output-Befehl im angegebenen Speicherort unter jenem Namen. Statt input ist auch das Kürzel *i*, sowie für output *o* möglich.

G'MIC bietet die Möglichkeit eigene Funktionen mit entsprechenden Befehlen zu erstellen. Hierfür wird eine Datei mit dem Namen *user.gmic* im Verzeichnis *C:\Users\Ron\AppData\Roaming* (kurz: *%APPDATA%*) platziert.

Im Rahmen der Masterarbeit sind drei Funktionen in dieser *user.gmic*-Datei definiert worden. „my\_oilpainting“, „my\_inkdrawing“ und „my\_silhouette“ enthalten die für den jeweiligen Workflow notwendigen Befehle. Hinter der Filterbezeichnung folgen die benutzten Parametereinstellungen. Den Inhalt zeigen Tabelle 4 und Tabelle 5 ausführlich.

Der *blend*-Befehl („Blend [Standard]“ unter Kategorie „Layers“) weist den Ebenenmodus zu und fast die beteiligten Ebenen, angegeben in eckigen Klammern, anschließend zusammen. Damit kann das beschriebene Ebenenmanagement über die CLI realisiert werden. Die Ebenen werden in Listen organisiert. Das erste Bild der Liste hat den Index 0 angesprochen durch [0]. Das zweite Bild hat den Index 1 angesprochen mit [1] und so weiter<sup>11</sup>. Der Befehl [0]x2 erstellt z.B. zwei Kopien der Ebene 0.

---

<sup>10</sup> <https://gmic.eu/download.shtml>

<sup>11</sup> <https://gmic.eu/reference.shtml>

Zeile	Code
1	
2	my_oilpainting :
3	
4	gcd_normalize_brightness 0,10,0,3,0,0
5	fx_retinex 75,16,1,1,1,5,15,80,250,0,50,50
6	fx_tk_colortemp 10,0,0
7	fx_equalize_local_histograms 75,2,4,100,8,1,16,0,50,50
8	fx_freaky_details 2,10,1,11,0,32,0,50,50
9	fx_deblur 2,10,20,0.1,1,11,0,24,0,50,50
10	fx_posterize 150,30,1,12,0,0,0,0,50,50
11	[0]x1
12	fx_engrave[0] 1,15,2,2,70,0,0,1,5,1,0,15,0,1,0
13	fx_blend[0,1] 26,0,100,1,"1/2 - 1/4*cos(pi*a) - 1/4*cos(pi*b)"
14	fx_sketchbw 3,45,180,30,1.75,0.02,0.5,0.75,0.1,0.7,3,6,0,1,4,0,50,50
15	[0]x2
16	fx_blur_bloom[0] 1,2,5,0,0,0,0,0,7,0,50,50
17	fx_blend[0,1] 14,0,100,1,"1/2 - 1/4*cos(pi*a) - 1/4*cos(pi*b)"
18	fx_blackandwhite[0] 0.3,0,0.59,0,0.11,0,0,0,0,0,0,0,0,2,0,0,0,16,4,0,0,0,50,50
19	fx_blend[0,1] 5,0,100,1,"1/2 - 1/4*cos(pi*a) - 1/4*cos(pi*b)"
20	fx_smooth_anisotropic 60,0.9,0.64,3.1,1.1,0.8,30,2,0,1,1,0,0,24
21	fx_stroke 12,50,0,2,1,100,0,0,255,255,255,255,255,255,255,255,0,0,0,0,0,0,0,1,0 gui_merge_layers
22	fx_drop_shadow 3,3,1.8,0,0,0
23	
24	

Tabelle 4: Inhalt Datei user.gmic

Zeile	Code
25	my_inkdrawing :
26	
27	gcd_normalize_brightness 0,10,0,3,0,0
28	fx_retinex 75,16,1,1,1,5,15,80,250,0,50,50
29	fx_tk_colortemp 10,0,0
30	fx_equalize_local_histograms 75,2,4,100,8,1,16,0,50,50
31	fx_freaky_details 2,10,1,11,0,32,0,50,50
32	fx_deblur 2,10,20,0.1,1,11,0,24,0,50,50
33	fx_blackandwhite 0.3,0,0.59,0,0.11,0,0,0,0,0,0,0,0,0,2,0,0,0,16,4,0,0,0,50,50
34	fx_morphological 5,0,4,"1,0,1; 0,1,0; 1,0,1",1,0,0,0,0,50,50
35	fx_stamp 1,50,0,0,0,0,1,0,50,50
36	fx_smooth_antialias 5,10,0.8,0,50,50
37	fx_stroke 12,50,0,2,1,100,0,0,255,255,255,255,255,255,255,255,0,0,0,0,0,0,0,1,0 gui_merge_layers
38	fx_drop_shadow 3,3,1.8,0,0,0
39	
40	
41	my_silhouette :
42	
43	fx_adjust_colors -100,-100,0,0,0,0,50,50
44	fx_stroke 12,50,0,2,1,100,0,0,255,255,255,255,255,255,255,255,0,0,0,0,0,0,0,1,0 gui_merge_layers
45	fx_drop_shadow 3,3,1.8,0,0,0
46	

Tabelle 5: Fortsetzung Inhalt Datei user.gmic

Das oben genannte Muster anwendend, lautet die Anweisung für den automatisierten Ablauf des „Ölmalerei“-Workflows über die Konsole beispielhaft wie folgt:

```
gmic.exe -i D:\Workflow\Bilder\Frauenkirche.png -my_oilpainting -o D:\Workflow\Ausgabe\Frauenkirche_Oelmalerei.png
```

Beispielanweisung für Workflow „Tuschezeichnung“:

```
gmic.exe -i D:\Workflow\Bilder\Frauenkirche.png -my_inkdrawing -o D:\Workflow\Ausgabe\Frauenkirche_Tuschezeichnung.png
```

Und zu guter Letzt ein Beispiel für den Workflow „Silhouette“:

```
gmic.exe -i D:\Workflow\Bilder\Frauenkirche.png -my_silhouette -o D:\Workflow\Ausgabe\Frauenkirche_Silhouette.png
```

Diesem Prinzip folgend können die Befehle den zu Grunde liegenden Ausgangsbildern entsprechend angepasst und so der Vorgang automatisiert abgelaufen werden.

## 4.3 Anwendung: Karte Dresden

Die praktische Arbeit schließt eine Kartendarstellung ab. Diese präsentiert die Ergebnisse aus den Durchläufen der automatisierten Workflows. Sie sollen folgend Maßstabsbereiche definieren.

Als Datengrundlage wurde eine OSM-Karte verwendet und in QGIS entsprechend aufbereitet. In einem zum Projekt hinzugefügten Punktelayer wurden insgesamt 9 prominente Gebäude in Dresden ausgewählt. Diese sind Frauenkirche, Hofkirche, Kronentor, Residenzschloss, Semperoper, Kulturpalast, Goldener Reiter, Yenidze und Rudolf-Harbig-Stadion. Über die Layereigenschaften wurde eine regelbasierte Symbolisierung der gesetzten Punkte vorgenommen. Die Ölmalerei-Signatur ist dabei im Maßstabsbereich von 1:1.000 bis 1:9.999, die Tusche-Signatur von 1:10.000 bis 1:19.999 und die Silhouetten-Signatur von 1:20.000 bis 1:99.999 sichtbar. Eine Beschriftung mit den Namen aus der Attributtabelle wurde ebenfalls regelbasiert hinzugefügt. Die Ergebnisse sind folgend in Kartenausschnitten dargestellt (Abbildung 30, Abbildung 31 und Abbildung 32).

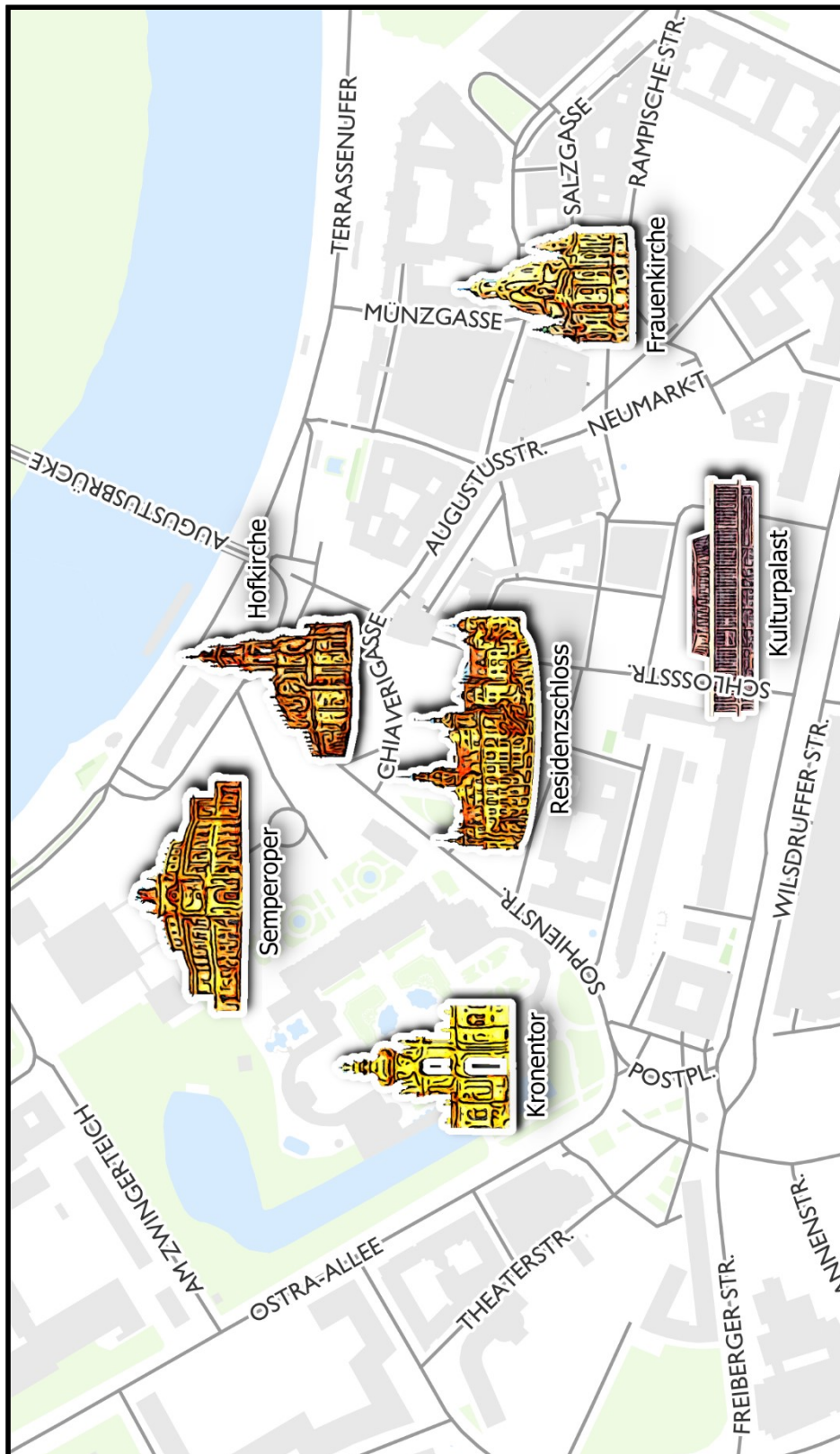


Abbildung 30: Ausschnitt Karte Dresden mit Ölmalerei-Signaturen (Maßstab 1:8.000)



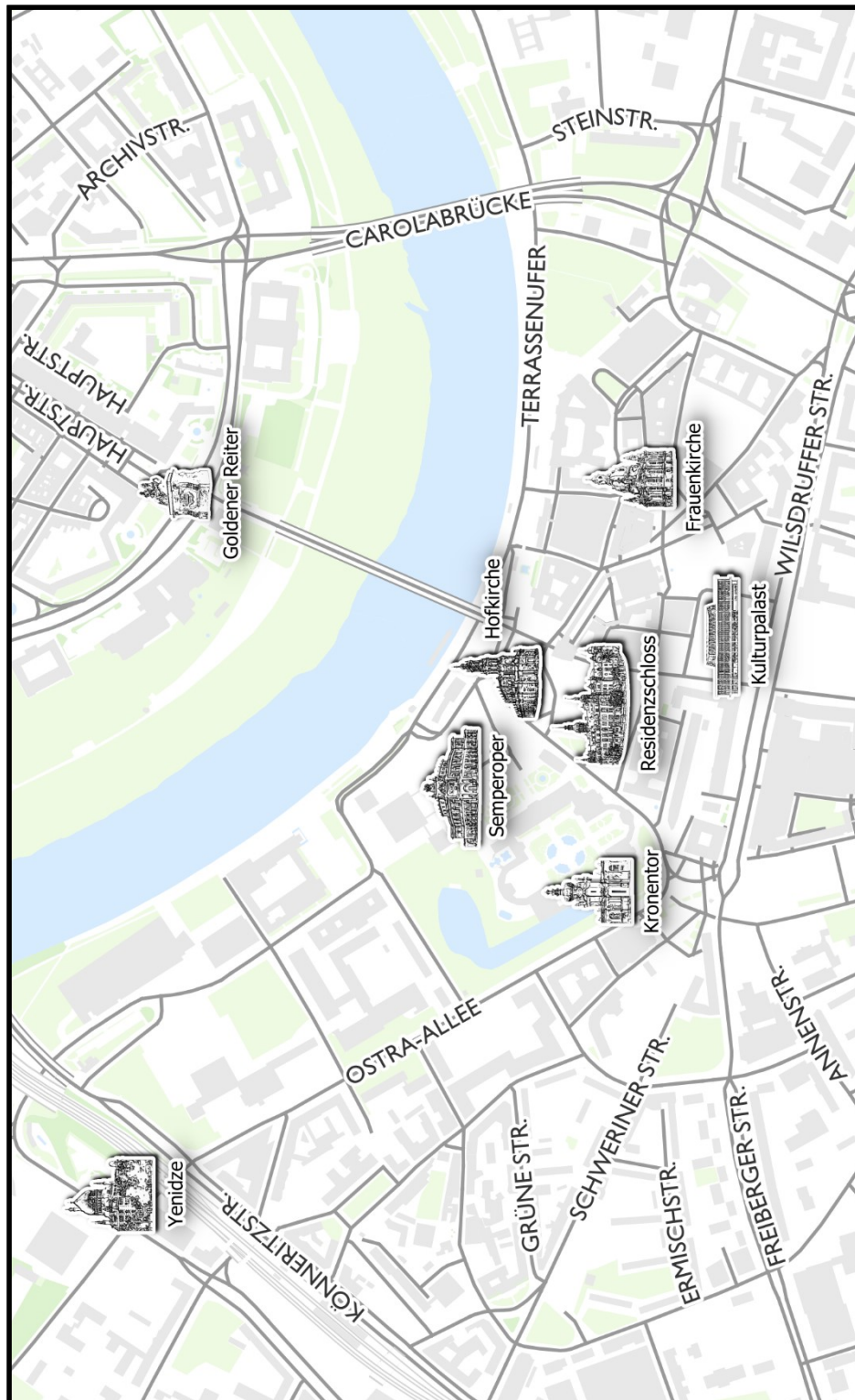


Abbildung 31: Ausschnitt Karte Dresden mit Tuschezeichnung-Signaturen (Maßstab 1:16.000)

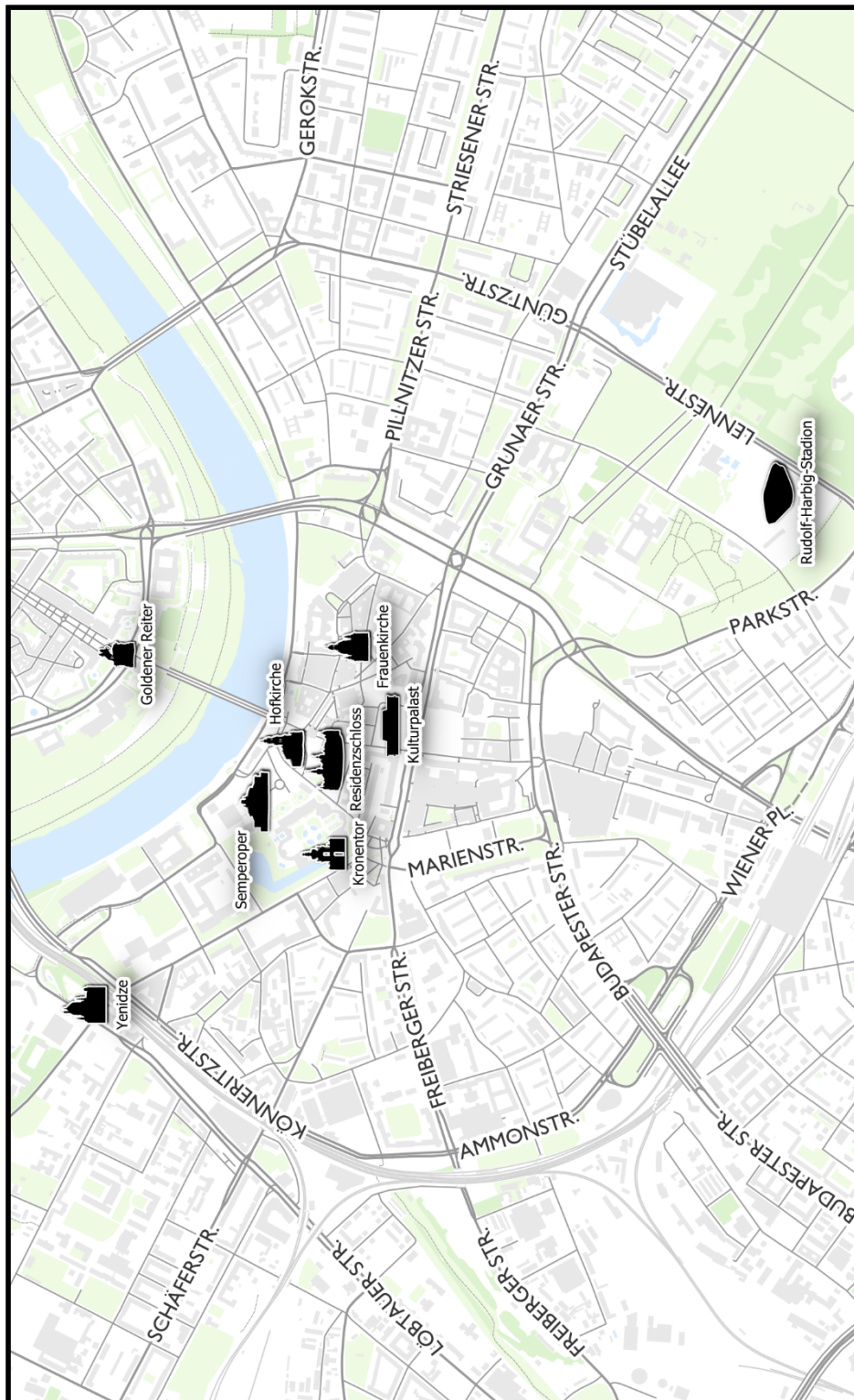


Abbildung 32: Ausschnitt Karte Dresden mit Silhouette-Signaturen (Maßstab 1:32.000)

## 4.4 Neural Style Transfer

In Kapitel 2.1 wurde die Theorie zum Neural Style Transfer (NST) vorgestellt. An dieser Stelle soll nun die praktische Nutzung des Konzepts anhand konkreter Anwendungen und deren Ergebnisse gezeigt werden.

### 4.4.1 Online-Anwendungen

NST ist nicht nur Forschungsschwerpunkt auf akademischer Ebene, sondern findet auch großen Anklang in der Industrie. DeepArt.io<sup>12</sup> bietet die Möglichkeit ein Foto hochzuladen und einen vorgegebenen Stil auszuwählen oder einen eigenen hochzuladen. Innerhalb weniger Minuten wird das Ergebnis erstellt und zum Download angeboten. Eine ähnliche Anwendung, die auf DeepArt beruht, ist die App Prisma<sup>13</sup>, die es ermöglicht ein mit dem Smartphone aufgenommenes Foto sofort in ein Kunstwerk zu verwandeln.

Ein weiteres Beispiel ist die von DeepArtEffects<sup>14</sup> entwickelte Deep Art Effects API, die auf einer Beispiel-Webseite<sup>15</sup> (Abbildung 33) getestet werden kann. Die Ergebnisse sind recht imposant. Allerdings besteht hier nicht die Möglichkeit des Uploads eines eigenen Stils und auch die Parameter zur Berechnung des NST können nicht beeinflusst werden. Die Implementierung ist daher eine Black Box.

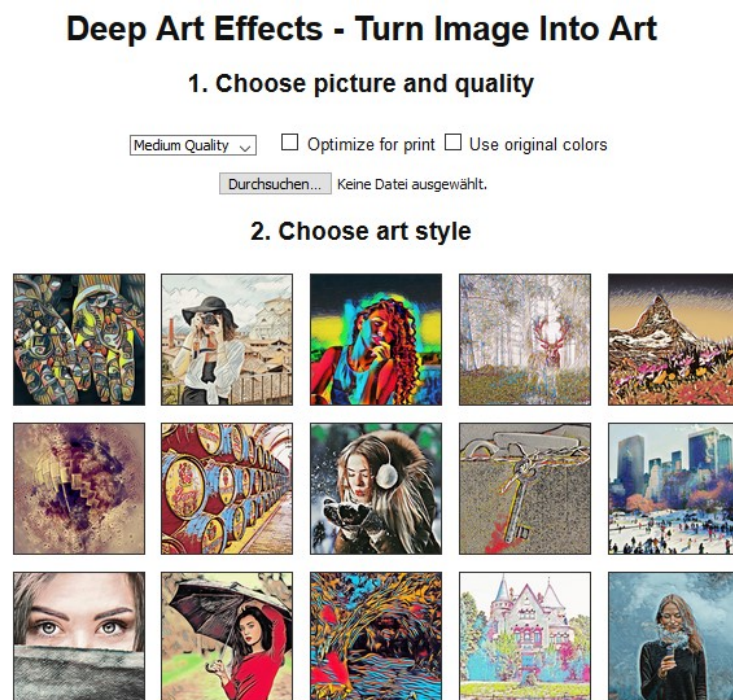


Abbildung 33: Showcase Webseite von Deep Art Effects Developer

<sup>12</sup> <https://deepart.io/>

<sup>13</sup> <https://prisma-ai.com/>

<sup>14</sup> <https://developer.deeparteffects.com/>

<sup>15</sup> <http://showcase.deeparteffects.com/>





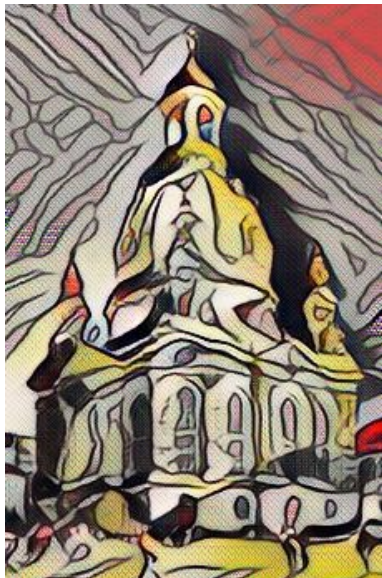






Inputbild	Stil	Output
		
		
		

Abbildung 34: Beispiele NST über Webseite von Deep Art Effects Developer

#### 4.4.2 Offline-Implementierung

Für einen intensiven praktischen Test zum Thema Neural Style Transfer wurde die TensorFlow<sup>16</sup> Implementierung von (Smith, 2016) verwendet. Der Anwendung liegen jene bereits vorgestellten Techniken zu Grunde, die in den erwähnten Arbeiten von (Gatys, Ecker, & Bethge, 2016) und (Gatys, Bethge, Hertzmann, & Shechtman, 2016) beschrieben sind und basiert außerdem auf der Torch<sup>17</sup> Implementierung von (Johnson, 2015). Zusätzlich werden Techniken für die semantische Segmentierung und die Übertragung mehrerer Stile bereitgestellt.

Die Anwendung bietet eine Vielzahl von Einstellmöglichkeiten, um Einfluss auf das Ergebnis des NST zu nehmen. Im Rahmen dieser Arbeit bestand das Ziel darin den in Kapitel 4.1.1 entwickelten Ölmalerei-Workflow zunächst auf ein Bild wie dort beschrieben anzuwenden und das Ergebnis in Folge als Stilbild für den Neural Style Transfer-Prozess zu verwenden. Abbildung 35 zeigt den verwendeten Ausschnitt des oft zitierten Frauenkirche-Bildes als Quelle (links) und rechts das Renderingergebnis durch das G'MIC-CLI (Einzelheiten in Kapitel 4.2).



Abbildung 35: Ausgangsbild Frauenkirche (links) und Ergebnis Workflow Ölmalerei (rechts)

Die Implementierung wird über die Konsole ausgeführt. Einzelheiten zu Voraussetzungen siehe (Smith, 2016). Im Theorieteil ist es bereits angesprochen worden, dass mittels eines Convolutional Neural Networks es möglich ist Inhalt und Stil gut zu separieren, um dann beide Repräsentationen unabhängig voneinander zu manipulieren und schließlich aussagekräftige künstlerische Bilder zu generieren (Gatys, Ecker, & Bethge, 2016). Um diesen Umstand zu demonstrieren wurde das in Abbildung 35 (rechts) erzeugte Workflow-Ergebnis als Stilbild definiert und auf insgesamt 10 Bilder

<sup>16</sup> <https://www.tensorflow.org/>

<sup>17</sup> <http://torch.ch/>



bekannter Sehenswürdigkeiten in Deutschland angewendet. Die Ergebnisse sind in Tabelle 7 zu sehen. Es wird jeweils das verwendete Ausgangsbild (Input) und das künstlerische Ergebnis nach dem Stiltransfer gegenübergestellt. (Gatys, Ecker, & Bethge, 2016) verwenden für die Inhaltsdarstellung den Layer 'conv4\_2' und für die Stilerzeugung mehrere Layer aus dem CNN ('conv1\_1', 'conv2\_1', 'conv3\_1', 'conv4\_1' und 'conv5\_1'). Das heißt, aus jeder Faltungsschicht des Netzwerkes wird ein Layer zur Stilkonstruktion verwendet. (Smith, 2016) benutzt die Rectified Linear Unit (ReLU). Eine Aktivierungsfunktion, die dafür sorgt, dass alle negativen Werte im Bild auf Null und alle Werte die größer als Null sind unverändert bleiben<sup>18</sup>.

Tabelle 6 zeigt die wichtigsten Parameter noch einmal zusammengefasst. Die restlichen, an der Umsetzung beteiligten Größen wurden mit den Default-Werten verwendet.

Parameter	Wert
<b>content_layers</b>	'conv4_2'
<b>style_layers</b>	'relu1_1', 'relu2_1', 'relu3_1', 'relu4_1', 'relu5_1'
<b>style_layer_weights</b>	0.2 für jeden beteiligten Layer, die anderen 0
<b>content_weight</b>	7.5e1
<b>style_weight</b>	1e4
<b>max_iterations</b>	500
<b>max_size</b>	512px

Tabelle 6: Übersicht der wichtigsten Parameter und verwendete Werte

Schlussendlich wurden die Ergebnisse in einer Karte zusammengefasst (Abbildung 37). Sie zeigt die ausgeschnittenen Objekte platziert auf einem Kartenbeispiel. Die Karte selbst ist eine OSM-Base-map im Maßstab 1:10 Mio. und zeigt Deutschland. Zum Test wurde diese zunächst ebenfalls als Bild exportiert und mit einem Stilbild (Vincent van Goghs „Sternennacht“) gerendert, um auch hier eine stilistische künstlerische Anpassung zu generieren (Abbildung 36). Die Inspiration hierfür stammt aus dem Paper von (Bogucka & Meng, 2019). Folgende Parameter wurden dafür verwendet: Inhaltsgewichtung: 1e1, Stilgewichtung 1e1, Iterationen: 100 und max. Größe: 535px. Abschließend wurde in GIMP das Arrangement der Icons vorgenommen und exportiert.

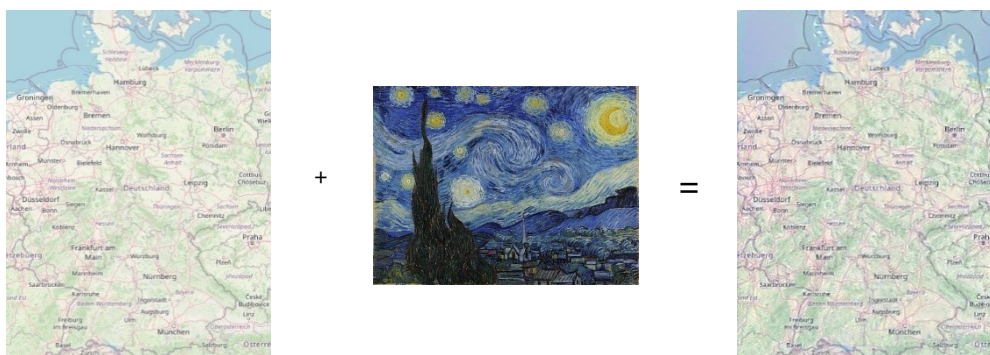


Abbildung 36: NST Basemap

<sup>18</sup> <https://jaai.de/convolutional-neural-networks-cnn-aufbau-funktion-und-anwendungsgebiete-1691/>





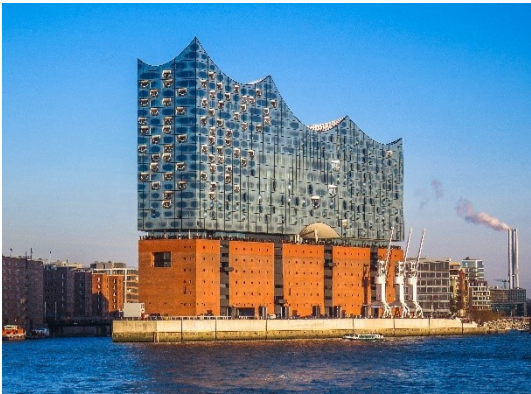



Input	Output
	
	
	
	









Tabelle 7: Übersicht der Ergebnisse des NST

Alle Bilder unter Creative Commons-Lizenz, von oben nach unten: Basteibrücke, Brandenburger Tor, Elbphilharmonie Hamburg, Zeche-Zollverein Essen, Frauenkirche München, Holstentor Lübeck, Kölner Dom und Hohenzollernbrücke, Schloss Neuschwanstein<sup>19</sup>, Skyline Frankfurt am Main, Völkerschlachtdenkmal bei Leipzig [eigene Aufnahme]).

<sup>19</sup> Aufnahme von Tourist Information Schwangau  
(<https://www.schwangau.de/koeniglich/schloss-neuschwanstein/>)



Abbildung 37: Ergebnis NST - Karte Deutschland mit Icons der Sehenswürdigkeiten

## 5 Diskussion

### 5.1 Resultate

#### 5.1.1 Bildverarbeitung

Analysiert man die Ergebnisse der Karten aus Abbildung 30, Abbildung 31 und Abbildung 32 lässt sich klar feststellen, dass die Erzeugung ikonographischer Symbole durch Bildverarbeitungsmethoden semi-automatisiert möglich ist. Allerdings gibt es sowohl Stärken als auch Schwächen bei der Umsetzung.

Der Ölmalerei-Workflow wirkt nach langer Entwicklungsphase im kleinen Format endlich künstlerisch. Einzig zu kritisieren sind die Farbunterschiede mancher Resultate. Diese rühren daher, dass für alle Objekte dieselben Einstellparameter verwendet wurden. Nur durch eine händische Bearbeitung (siehe Kapitel 4.1) und damit objektspezifischen Wertanpassung in den entsprechenden Filtern (siehe Tabelle 3) kann das Ergebnis verbessert werden.

Natürlich spielt auch das Ausgangsfoto eine entscheidende Rolle. Ist ein Bild besonders hell oder dunkel, müsste ggf. im „Normalize Brightness“-Filter nachjustiert werden. Aber im Automatisierungsprozess ist das nicht möglich. Die Größe wurde bereits angesprochen. Bei der vorliegenden Entwicklung sollte das Bild nicht höher als 300px sein, was ebenfalls Ergebnisentscheidend ist.

Aufgrund der Individualität der Workflows wurde darauf verzichtet, Dinge aus dem Hintergrund bzw. der Umgebung den Objekten beizufügen und diese stattdessen komplett freizustellen. Es werden so Fehlinterpretationen minimiert und die Platzierung auf der Karte erleichtert. In diesem Zusammenhang wurde auch die Notwendigkeit eines Halo-Effektes untersucht. Entschieden wurde sich für eine Konturlinie mit Schatten. Dies bewirkt eine Abhebung gegenüber der Umgebung und verleiht eine gewisse Plastizität.

Die Ergebnisse aus dem Workflow „Silhouette“ zeigen sehr gut, dass es aber auch problematische Objekte gibt, die sich wenig bis gar nicht für diese Darstellung eignen. Schaut man sich z.B. den Kulturpalast und das Rudolf-Harbig-Stadion an, so fällt die Interpretation relativ schwer. Ohne Beschriftung erst recht. An dieser Stelle müsste evtl. in GIMP nachgearbeitet werden, um die Graphik deutbarer zu gestalten. Allerdings ist das sehr schwierig, gerade in dieser Größenordnung. Hier stößt der Einsatz der Darstellung an seine Grenzen. Das ist auch der Grund, warum hauptsächlich Front-View Ansichten der Sehenswürdigkeiten verwendet wurden.

Für den praktischen Einsatz z.B. in einer Touristenkarte ist auf jeden Fall die Ölmalerei-Darstellung zu empfehlen. Die Tuschezeichnungen haben Potential, vor allem in einem größeren Format. Auf der Karte würde man sich teils etwas ausgedünnte Strukturen wünschen. Vergleich hierzu Kapitel 4.1.2 und die dargestellten Ergebnisse von Frauenkirche und Hofkirche. Aber dies kann nur durch eine spezifische Bearbeitung erreicht werden.

#### 5.1.2 Neural Style Transfer

Mit dem Kartenbeispiel (Abbildung 37) ist das Potential des Neural Style Transfers für die Kartographie deutlich geworden. Durch die Wahl eines geeigneten Stilbilds ist es möglich ansprechende ikonische Kartenzeichen prominenter Gebäude zu erzeugen. Das Resultat ist vergleichbar mit den



Ergebnissen aus der Bildverarbeitung mit entsprechendem Filtering. Lediglich das Finden geeigneter Stile, sowie die Einstellung der Parameter, wie die Gewichtung von Inhalt und Stil sind Faktoren die diskutiert und teilweise intensiv ausprobiert werden müssen. Nicht jedes Stilbild ist für jeden Bearbeitungsprozess geeignet. Außerdem gibt es auch an dieser Stelle kritische Objekte, wie z.B. Kulturpalast (Abbildung 38) oder auch Schloss Neuschwanstein auf der Karte, wo die Deutung des Ergebnisses schwer fällt. Für sich betrachtet ist es Kunst, aber zum Einsatz auf einer Karte und dem möglichen Erkennungswert fraglich. In der Arbeit wurde explizit der Ölmalerei-Stil ausgewählt, um die Leistungsfähigkeit der Implementierung zu testen. Weiteres Potential bieten außerdem die diversen Einstellmöglichkeiten. Farberhaltung (Gatys, Bethge, Hertzmann, & Shechtman, 2016), sowie der Einsatz mehrerer Stile (Gatys L. A., Ecker, Bethge, Hertzmann, & Shechtman, 2017) sind nur zwei interessante Themen, die das Ergebnis noch ansprechender gestalten können und weiterer Untersuchungen bedürfen.



Abbildung 38: Kulturpalast Dresden - NST mit Ölmalereistil

## 5.2 Ausblick

Interessant für weitere Betrachtungen ist die künstlerische Abstraktion der Hintergrundkarte ebenfalls durch einen angepassten Workflow. (Dong, 2015) und (Hartl, 2015) behandeln in ihren Arbeiten entsprechende Ansätze für Webkarten. Die evtl. Kombination von beiden Ansätzen wäre wünschenswert.

Ebenso spielen 3D-Stadtmodelle heute eine wichtige Rolle. Inwieweit kann hier so ein künstlerisch sehr abstrahierter Workflow eingesetzt werden.

## 6 Zusammenfassung

Mit der praktischen Arbeit konnte gezeigt werden, dass es möglich ist bildhafte Signaturen aus Aufnahmen beliebiger Herkunft mit Bildbearbeitungstools zu erzeugen. Außerdem ist es gelungen den Arbeitsprozess teilweise zu automatisieren. Drei Workflows sind entstanden. Die Ergebnisse „Ölmalerei“, „Tuschezeichnung“ und „Silhouette“ repräsentieren schließlich drei Maßstabsbereiche auf einer Karte. Außerdem wurde die Theorie des Neural Style Transfer behandelt und das Potential der Anwendung ebenfalls an einem Kartenbeispiel demonstriert.

## Literaturverzeichnis

- Bogucka, E. P., & Meng, L. (2019). *Projecting emotions from artworks to maps using neural style transfer*. Proceedings of the ICA, Volume 2, 2019, id.9. doi:10.5194/ica-proc-2-9-2019
- DeCarlo, D., & Santella, A. (2002). *Stylization and Abstraction of Photographs*. ACM Transactions on Graphics, July 2002. doi:10.1145/566654.566650
- Desclaux-Salachas, J. (2017). *Cartographics - Die Kunst der Kartengestaltung*. München: Prestel Verlag.
- Dev, K. (2013). Mobile Expressive Renderings: The State of the Art. *IEEE Computer Graphics and Applications (Volume 33, Issue 3, May 2013)*, (S. 22-31). doi:10.1109/MCG.2013.20
- Dong, L. (2015). *The Derivation of Digital Embroid Styled Maps*. TU Dresden, Masterarbeit.
- Gatys, L. A., Bethge, M., Hertzmann, A., & Shechtman, E. (2016). *Preserving Color in Neural Artistic Style Transfer*. arXiv:1606.05897 [cs.CV].
- Gatys, L. A., Ecker, A. S., & Bethge, M. (2015). *A Neural Algorithm of Artistic Style*. arXiv:1508.06576v2 [cs.CV].
- Gatys, L. A., Ecker, A. S., Bethge, M., Hertzmann, A., & Shechtman, E. (2017). *Controlling Perceptual Factors in Neural Style Transfer*. arXiv:1611.07865v2 [cs.CV].
- Gatys, L., Ecker, A. S., & Bethge, M. (2016). Image Style Transfer Using Convolutional Neural Networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (S. 2414-2423). Las Vegas. doi:10.1109/CVPR.2016.265
- Gooch, A. A. (2010). Towards mapping the field of Non-Photorealistic Rendering. *NPAR 2010, ACM*, (S. 159-164). doi:10.1145/1809939.1809958
- Hartl, M. K. (2015). *Design and implementation of an automated workflow to provide a zoomable web mapping application using artistic styles*. TU Dresden, Masterarbeit.
- Hertzmann, A. (2010). Non-Photorealistic Rendering and the Science of Art. *NPAR 2010, ACM*, (S. 147-157). doi:10.1145/1809939.1809957
- Jing, Y., Yang, Y., Feng, Z., Ye, J., Yu, Y., & Song, M. (2018). *Neural Style Transfer - A Review*. arXiv:1705.04058v7 [cs.CV].
- Johnson, J. (2015). *Torch implementation of neural style algorithm*. Von GitHub Repository: <https://github.com/jcjohnson/neural-style> abgerufen
- Johnson, J., Alahi, A., & Fei-Fei, L. (2016). *Perceptual Losses for Real-Time Style Transfer and Super-Resolution*. arXiv:1603.08155 [cs.CV].
- Kyprianidis, J. E., Collomosse, J., Wang, T., & Isenberg, T. (2013). State of the 'Art': A Taxonomy of Artistic Stylization Techniques for Images and Video. *IEEE Transactions on Visualization and Computer Graphics (Volume: 19, Issue: 5, May 2013)*, (S. 51-58). doi:10.1109/TVCG.2012.160

- Lansdown, J., & Schofield, S. (1995). Expressive Rendering: A Review of Non-photorealistic Techniques. *IEEE Computer Graphics and Applications (Volume: 15, Issue: 3, May 1995)*, (S. 29-37). doi:10.1109/38.376610
- Rosin, P., & Collomosse, J. (2013). Image and Video based Artistic Stylisation. *Computational Imaging and Vision*, Vol. 42, Springer, London/Heidelberg.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., . . . Fei-Fei, L. (2015). *ImageNet Large Scale Visual Recognition Challenge*. arXiv:1409.0575v3 [cs.CV].
- Semmo, A., Isenberg, T., & Döllner, J. (2017). *Neural Style Transfer: A Paradigm Shift for Image-based Artistic Rendering?* NPAR 2017. doi:10.1145/3092919.3092920
- Shirley, P., Marschner, S., Ashikhmin, M., Gleicher, M., Hoffman, N., Johnson, G., . . . Wyvill, B. (2009). *Fundamentals of Computer Graphics (Third Edition)*. CRC Press, Taylor & Francis Group.
- Simonyan, K., & Zisserman, A. (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv:1409.1556v6 [cs.CV].
- Smith, C. (2016). *TensorFlow (Python API) implementation of Neural Style*. Von GitHub Repository: <https://github.com/cysmith/neural-style-tf> abgerufen
- Strothotte, T., & Schlechtweg, S. (2002). *Non-Photorealistic Computer Graphics: Modeling, Rendering, and Animation*. San Francisco: Elsevier Science.
- Winkenbach, G., & Salesin, D. H. (1994). *Computer-Generated Pen-and-Ink Illustration*. SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques (July 1994). doi:10.1145/192161.192184
- Winnemöller, H. (2013). NPR in the Wild. In P. Rosin, & J. Collomosse, *Image and Video-Based Artistic Stylisation* (S. 353-374). *Computational Imaging and Vision*, Vol. 42, Springer, London/Heidelberg.
- Wolf, J. (2012). *Gimp 2.8 - Das umfassende Handbuch* (1. Ausg.). Bonn: Galileo Press.
- Wood, J., Isenberg, P., Isenberg, T., Dykes, J., Boukhelifa, N., & Slingsby, A. (2012). *Sketchy Rendering for Information Visualization*. *IEEE Transactions on Visualization and Computer Graphics* (Volume: 18, Issue: 12, Dec. 2012). doi:10.1109/TVCG.2012.262
- Yao, C., Li, Y., & Qi, Y. (2019). *Research on Neural Style Transfer Algorithm*. IOP Conference Series: Materials Science and Engineering 569 052107. doi:10.1088/1757-899X/569/5/052107